

The Information Retrieval Series

Mihai Lupu

Katja Mayer

John Tait

Anthony J. Trippe

Current Contents  
in Patent

*Series Editor*

W. Bruce Croft

*Editorial Board*

ChengXiang Zhai

Maarten de Rijke

Nicholas J. Belkin

Charles Clarke

Mihai Lupu • Katja Mayer • John Tait •  
Anthony J. Trippe

Editors

# Current Challenges in Patent Information Retrieval

 Springer

*Editors*

Mihai Lupu  
Information Retrieval Facility  
Donau-City Straße 1  
Vienna 1220  
Austria  
[m.lupu@ir-facility.org](mailto:m.lupu@ir-facility.org)

John Tait  
Information Retrieval Facility  
Donau-City Straße 1  
Vienna 1220  
Austria  
[john.tait@ir-facility.org](mailto:john.tait@ir-facility.org)

Katja Mayer  
Information Retrieval Facility  
Donau-City Straße 1  
Vienna 1220  
Austria  
[k.mayer@ir-facility.org](mailto:k.mayer@ir-facility.org)

Anthony J. Trippe  
3LP Advisors  
Post Rd. 7003 Suite 409  
43016 Dublin, OH  
USA  
[tony@trippe.com](mailto:tony@trippe.com)

ISSN 1387-5264

ISBN 978-3-642-19230-2

e-ISBN 978-3-642-19231-9

DOI 10.1007/978-3-642-19231-9

Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011926006

ACM Computing Classification (1998): H.3, I.7, J.1

© Springer-Verlag Berlin Heidelberg 2011

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

*Cover design:* deblik

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))

# Preface

Patent Information Retrieval is an economically important activity. Today's economy is becoming increasingly knowledge-based and intellectual property in the form of patents plays a vital role in this growth. Between 1998 and 2008, the number of patent applications filed worldwide grew by more than 50 percent. The number of granted patents worldwide continues to increase, albeit at a slower rate than at its peak in 2006 (18%), when some 727,000 patents were granted. The substantial increase in patents granted is due, in part, to efforts by patent offices to reduce backlogs as well as the significant growth in the number of patents granted by China and, to a lesser extent in the more recent years, by the Republic of Korea. According to these statistics, the total number of patents in force worldwide at the end of 2008 was approximately 6.7 million (WIPO report 2010). A prior art search might have to cover as many as 70 million patents. By combining data from Ocean Tomo's Intangible Asset Market Value Survey, and Standard and Poor's 1200 Index we can estimate that the global value of patents exceeds US\$10 trillion in 2009.

A patent is a bargain between the inventor and the state. The inventor must teach the community how to make the product, and use the techniques he/she has invented in return for a limited monopoly which gives him a set time to exploit his invention and realise its value. Patents are used for many reasons, e.g. to protect inventions, to create value and to monitor competitive activities in a field. Much knowledge is distilled through patents, which is never published elsewhere. Thus patents form an important knowledge resource—e.g. much technical information represented in patents is not represented in scientific literature—and are at the same time important legal documents.

Despite the overall increase in patent applications and grants, a situation of economic downturn, such as the one the world has experienced in 2008, leads to a reduction in patent applications and grants (as indicated by preliminary figures published by WIPO for 2009). This is, to some extent, explained by the high costs involved in applying for a patent, particularly for small enterprises. The costs of the pre-application process, the long duration of the application process and the corresponding uncertainty in the long-term economy in such periods of economic downturn need to be addressed by changing the way we search the patent and non-patent

literature. Both the Intellectual Property (IP) professionals and the Information Retrieval (IR) scientists can see this book as a challenge: for the former, in terms of adapting to new tools; for the latter, in terms of creating better tools for an obviously difficult task; for both, in terms of engaging in exchange and cooperation.

In the past 10 or 15 years, general information retrieval and Web search engines have made tremendous advances. And still, we see a huge gap between the technologies which, on the one hand, were emerging from research labs and in use by major internet search engines, in e-commerce, and in enterprise search systems, and, on the other, the systems in day-to-day use by the patent search communities.

It has been estimated that since 1991, when the US Federal National Institute of Standards and Technology (NIST) began its Text Retrieval Conference (TREC) evaluation campaign, the available information retrieval and search systems have improved 40% or more in their ability to find relevant documents. And yet the technologies underlying the patent search system were largely unaffected by these changes. Patent searchers generally use the same technology as in the 1980s. Boolean specification of searches and set-based retrieval are the norm rather than the ranked retrieval systems used by Google and the like. Tools in some areas have moved on significantly: some providers have semantic analysis tools, others effective visualisation mechanisms for patent documents. And yet there has not been the kind of revolution in patent search which Google had represented for Web search.

In the past few years, the Information Retrieval Facility (a not-for-profit research institution based in Vienna, Austria) has organised a series of events to bring together leading researchers in IR with those who practice and use patent search, to establish the interdisciplinary dialogue between the IR and the IP communities and to create a discursive as well as empirical space for sustainable discussion and innovation.

In the first Information Retrieval Facility Symposium in Vienna in 2007 ([www.irfs.at](http://www.irfs.at)), a distinguished audience of information retrieval scientists and patent search specialists started to explore the reasons for the knowledge gap. It turned out that academic researchers were often unaware of the specialised needs of the patent searchers: for example, they needed a degree of transparency quite unlike the casual Web searchers, upon which the academics mainly focussed. The patent searchers were often unaware of the advances made in other areas, and how they had been achieved. There were difficulties in finding (and using) a common, comprehensible vocabulary. In the course of that first Symposium, and through subsequent IRF symposia and other joint activities, such as the CLEF-IP and TREC-CHEM tracks, the PaIR and Aspire workshops, major progress has been made in developing a common understanding, and even an agenda between search researchers and technologists and the patent search community.

This book is part of the development of that joint understanding. Its origins lie in the idea of producing post-proceedings for the first IRF Symposium. That idea was not fully followed up, in part because of pressure to produce more practical, action-oriented work, and in part because many of the participants felt their approaches were at too early a stage for formal publication. In the course of the following years it became apparent there really was a demand to produce a volume which was accessible to both the patent search community and to the information retrieval research

community; to provide a collected and organized introduction to the work and views of the two sides of the emerging patent search research and innovation community; and to provide a coherent and organised view of what has been achieved and, perhaps even more significantly, of what remains to be achieved.

We have already noted the need for transparency (or at least defensibility) of search processes from the patent search community. We hope this book will allow the IR researchers to better understand why such transparency is needed, and what it means in practise. Furthermore, it is our hope that this book will also be a valuable resource for IP professionals in learning about current approaches of IR in the patent domain. It has often been difficult to reconcile the focus on useful technological innovation from the IP community, with the demands for scientific rigour and to proceed on the basis of sound empirical evidence, which is such an important feature of IR (in contrast to some other areas of computer science).

Moreover, patent search is an inherently multilingual and multinational topic: the novelty of a patent may be dismissed by finding a document describing the same idea in any language anywhere in the world. Patents are complex legal documents, even less accessible than the scientific literature. These are just some of the characteristics of the patent system, which make it an important challenge for the search, information retrieval and information access communities.

The book has had a lengthy and difficult gestation: the list of authors has been revised many times as a result of changes in institutional, occupational and private circumstances. Although we, the editors, do feel we have succeeded in producing a volume which will provide important perspectives of the issues affecting patent search research and innovation at the time of writing, as well as a useful, brief introduction to the outlook and literature of the community accessible to its members, regardless of their background, we would have liked to cover several topics not represented here.

In particular it was disappointing we could not include a chapter on NTCIR, the first of the evaluation campaigns to focus seriously on patents. Also, a chapter on the use of Latent Semantic Indexing for the patent domain had been planned, which ultimately could not appear in this book.

Several of the chapters have been written jointly by intellectual property and information retrieval experts. Members of both communities with a background opposite to the primary author have reviewed all the chapters. It has not always been easy to reconcile their differing viewpoints: we must thank them for taking the time to resolve their differences and for taking the opportunity to exchange their knowledge across fields and disciplinary mind-sets and to engage in a mutual discourse that will hopefully foster the understanding in the future.

Finally, we would like to thank the IRF for making this publication possible, the publisher, Springer; and in particular Ralf Gerstner, for the patience with which he accepted the numerous delays, as well as the external reviewers who read each chapter and provided the authors with valuable advice.

The editors are very grateful to the following persons, who agreed to review the manuscripts: Stephen Adams, Linda Andersson, Geetha Basappa, John M. Barnard, Shariq Bashir, Helmut Berger, Katrien Beuls, Ted Briscoe, Ben Carterette, Paul

Clough, Bruce Croft, Szabolcs Csepregi, Barrou Diallo, Karl A. Froeschl, Norbert Fuhr, Eric Gaussier, Julio Gonzalo, Allan Hanbury, Christopher G. Harris, Ilkka Havukkala, Bruce Hedin, Cornelis H.A. Koster, Mounia Lalmas, Patrice Lopez, Teresa Loughbrough, Marie-Francine Moens, Henning Müller, Iadh Ounis, Florina Piroi, Keith van Rijsbergen, Patrick Ruch, Philip Tetlow, Henk Thomas, Ingo Thon, Steve Tomlinson, Anthony Trippe, Suzan Verberne, Ellen M. Voorhees, Peter Willett, Christa Womser-Hacker.

Mihai Lupu  
Katja Mayer  
John Tait  
Anthony Trippe



# Contents

## Part I Introduction to Patent Searching

- 1 **Introduction to Patent Searching** . . . . . 3  
Doreen Alberts, Cynthia Barcelon Yang, Denise Fobare-DePonio, Ken Koubek, Suzanne Robins, Matthew Rodgers, Edlyn Simmons, and Dominic DeMarco
- 2 **An Introduction to Contemporary Search Technology** . . . . . 45  
Veronika Stefanov and John I. Tait

## Part II Evaluating Patent Retrieval

- 3 **Overview of Information Retrieval Evaluation** . . . . . 69  
Ben Carterette and Ellen M. Voorhees
- 4 **Evaluating Information Retrieval in the Intellectual Property Domain: The CLEF-IP Campaign** . . . . . 87  
Florina Piroi and Veronika Zenz
- 5 **Evaluation of Chemical Information Retrieval Tools** . . . . . 109  
Mihai Lupu, Jimmy Huang, and Jianhan Zhu
- 6 **Evaluating Real Patent Retrieval Effectiveness** . . . . . 125  
Anthony Trippe and Ian Ruthven

## Part III High Recall Search

- 7 **Measuring and Improving Access to the Corpus** . . . . . 147  
Richard Bache
- 8 **Measuring Effectiveness in the TREC Legal Track** . . . . . 167  
Stephen Tomlinson and Bruce Hedin

<b>9</b>	<b>Large-Scale Logical Retrieval: Technology for Semantic Modelling of Patent Search</b> . . . . .	181
	Hany Azzam, Iraklis A. Klampanos, and Thomas Roelleke	
<b>10</b>	<b>Patent Claim Decomposition for Improved Information Extraction</b> .	197
	Peter Parapatics and Michael Dittenbach	
<b>11</b>	<b>From Static Textual Display of Patents to Graphical Interactions</b> . .	217
	Steffen Koch and Harald Bosch	
<b>Part IV Classification</b>		
<b>12</b>	<b>Automated Patent Classification</b> . . . . .	239
	Karim Benzineb and Jacques Guyot	
<b>13</b>	<b>Phrase-based Document Categorization</b> . . . . .	263
	Cornelis H.A. Koster, Jean G. Beney, Suzan Verberne, and Merijn Vogel	
<b>14</b>	<b>Using Classification Code Hierarchies for Patent Prior Art Searches</b>	287
	Christopher G. Harris, Robert Arens, and Padmini Srinivasan	
<b>Part V Semantic Search</b>		
<b>15</b>	<b>Information Extraction and Semantic Annotation for Multi-Paradigm Information Management</b> . . . . .	307
	Hamish Cunningham, Valentin Tablan, Ian Roberts, Mark A. Greenwood, and Niraj Aswani	
<b>16</b>	<b>Intelligent Information Access from Scientific Papers</b> . . . . .	329
	Ted Briscoe, Karl Harrison, Andrew Naish, Andy Parker, Marek Rei, Advaith Siddharthan, David Sinclair, Mark Slater, and Rebecca Watson	
<b>17</b>	<b>Representation and Searching of Chemical-Structure Information in Patents</b> . . . . .	343
	John D. Holliday and Peter Willett	
<b>18</b>	<b>Offering New Insights by Harmonizing Patents, Taxonomies and Linked Data</b> . . . . .	357
	Andreas Pesenhofer, Helmut Berger, and Michael Dittenbach	
<b>19</b>	<b>Automatic Translation of Scholarly Terms into Patent Terms</b> . . . .	373
	Hidetsugu Nanba, Hideaki Kamaya, Toshiyuki Takezawa, Manabu Okumura, Akihiro Shinmori, and Hidekazu Tanigawa	
<b>20</b>	<b>Future Patent Search</b> . . . . .	389
	John I. Tait and Barou Diallo	
	<b>Index</b> . . . . .	409

# Contributors

**Doreen Alberts** Theravance Inc., 901 Gateway Blvd., South San Francisco, CA, USA

**Robert Arens** Nuance Communications, Burlington, MA, USA,  
[robert.arens@nuance.com](mailto:robert.arens@nuance.com)

**Niraj Aswani** Department of Computer Science, University of Sheffield, Sheffield, UK, [N.Aswani@dcs.shef.ac.uk](mailto:N.Aswani@dcs.shef.ac.uk)

**Hany Azzam** Queen Mary University of London, London, UK,  
[hany@eecs.qmul.ac.uk](mailto:hany@eecs.qmul.ac.uk)

**Richard Bache** Department of Computer and Information Sciences, University of Strathclyde, Glasgow G4 1XH, Scotland, UK, [richard.bache@gmail.com](mailto:richard.bache@gmail.com)

**Jean G. Beney** Dept. Informatique, LCI, INSA de Lyon, Lyon, France,  
[jean.beney@insa-lyon.fr](mailto:jean.beney@insa-lyon.fr)

**Karim Benzineb** SIMPLE SHIFT, Ruelle du P'tit-Gris 1, 1228 Plan-les-Ouates, Switzerland, [karim@simple-shift.com](mailto:karim@simple-shift.com)

**Helmut Berger** max.recall information systems, Vienna, Austria,  
[h.berger@max-recall.com](mailto:h.berger@max-recall.com)

**Harald Bosch** Institute for Interactive Systems and Visualization, Universität Stuttgart, Stuttgart, Germany

**Ted Briscoe** University of Cambridge, Cambridge, UK, [Ted.Briscoe@cl.cam.ac.uk](mailto:Ted.Briscoe@cl.cam.ac.uk); iLexIR Ltd, Cambridge, UK

**Ben Carterette** University of Delaware, Newark, DE 19716, USA,  
[carteret@cis.udel.edu](mailto:carteret@cis.udel.edu)

**Hamish Cunningham** Department of Computer Science, University of Sheffield, Sheffield, UK, [H.Cunningham@dcs.shef.ac.uk](mailto:H.Cunningham@dcs.shef.ac.uk)

**Dominic DeMarco** DeMarco Intellectual Property, LLC, 1111 16th Street, South Arlington, VA, USA

**Barou Diallo** European Patent Office, Patentlaan 2, 2288 EE Rijswijk Zh, Netherlands, [bdiallo@epo.org](mailto:bdiallo@epo.org)

**Michael Dittenbach** max.recall information systems, Vienna, Austria, [m.dittenbach@max-recall.com](mailto:m.dittenbach@max-recall.com)

**Denise Fobare-DePonio** Camarillo, CA, USA

**Mark A. Greenwood** Department of Computer Science, University of Sheffield, Sheffield, UK, [M.Greenwood@dcs.shef.ac.uk](mailto:M.Greenwood@dcs.shef.ac.uk)

**Jacques Guyot** SIMPLE SHIFT, Ruelle du P'tit-Gris 1, 1228 Plan-les-Ouates, Switzerland

**Christopher G. Harris** Informatics Program, The University of Iowa, Iowa City, IA, USA, [christopher-harris@uiowa.edu](mailto:christopher-harris@uiowa.edu)

**Karl Harrison** University of Cambridge, Cambridge, UK, [Harrison@hep.phy.cam.ac.uk](mailto:Harrison@hep.phy.cam.ac.uk)

**Bruce Hedin** H5, 71 Stevenson St., San Francisco, CA 94105, USA, [bhedin@h5.com](mailto:bhedin@h5.com)

**John D. Holliday** Information School, University of Sheffield, 211 Portobello Street, Sheffield S1 4DP, UK

**Jimmy Huang** York University, Toronto, Canada, [jhuang@yorku.ca](mailto:jhuang@yorku.ca)

**Hideaki Kamaya** Hiroshima City University, 3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima 731-3194, Japan, [kamaya@ls.info.hiroshima-cu.ac.jp](mailto:kamaya@ls.info.hiroshima-cu.ac.jp)

**Iraklis A. Klampanos** University of Glasgow, Glasgow, UK, [iraklis@dcs.gla.ac.uk](mailto:iraklis@dcs.gla.ac.uk)

**Steffen Koch** Institute for Interactive Systems and Visualization, Universität Stuttgart, Stuttgart, Germany

**Cornelis H.A. Koster** Computing Science Institute ICIS, Univ. of Nijmegen, Nijmegen, The Netherlands, [kees@cs.ru.nl](mailto:kees@cs.ru.nl)

**Ken Koubek** Koubek Information Consulting Services LLC, Wilmington, DE, USA

**Mihai Lupu** Information Retrieval Facility, Vienna, Austria, [m.lupu@ir-facility.org](mailto:m.lupu@ir-facility.org)

**Andrew Naish** Camtology Ltd, Cambridge, UK, [A.Naish@gmail.com](mailto:A.Naish@gmail.com)

**Hidetsugu Nanba** Hiroshima City University, 3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima 731-3194, Japan, [nanba@hiroshima-cu.ac.jp](mailto:nanba@hiroshima-cu.ac.jp)

**Manabu Okumura** Tokyo Institute of Technology, 4259 Nagatsuta-cho, Midori-ku, Yokohama 226-8503, Japan, [oku@pi.titech.ac.jp](mailto:oku@pi.titech.ac.jp)

**Peter Parapaties** Department of Software Technology and Interactive Systems, Vienna University of Technology, Favoritenstr. 9-11/188, 1040 Vienna, Austria, [p.parapaties@gmail.com](mailto:p.parapaties@gmail.com)

**Andy Parker** University of Cambridge, Cambridge, UK, [Parker@hep.phy.cam.ac.uk](mailto:Parker@hep.phy.cam.ac.uk); Camtology Ltd, Cambridge, UK

**Andreas Pesenhofer** max.recall information systems, Vienna, Austria, [a.pesenhofer@max-recall.com](mailto:a.pesenhofer@max-recall.com)

**Florina Piroi** Information Retrieval Facility, Vienna, Austria, [f.piroi@ir-facility.org](mailto:f.piroi@ir-facility.org)

**Marek Rei** University of Cambridge, Cambridge, UK, [Marek.Rei@hep.phy.cam.ac.uk](mailto:Marek.Rei@hep.phy.cam.ac.uk)

**Ian Roberts** Department of Computer Science, University of Sheffield, Sheffield, UK, [I.Roberts@dcs.shef.ac.uk](mailto:I.Roberts@dcs.shef.ac.uk)

**Suzanne Robins** Patent Information Services, Inc., Westborough, MA, USA

**Matthew Rodgers** Landon IP, Alexandria, VA, USA

**Thomas Roelleke** Queen Mary University of London, London, UK, [thor@eecs.qmul.ac.uk](mailto:thor@eecs.qmul.ac.uk)

**Ian Ruthven** Department of Computer and Information Sciences, University of Strathclyde, Glasgow, G12 8DY, UK, [ir@cis.strath.ac.uk](mailto:ir@cis.strath.ac.uk)

**Akihiro Shinmori** INTEC Systems Institute Inc., 1-3-3 Shinsuna, Koto-ku, Tokyo 136-0075, Japan, [shinmori\\_akihiro@intec-si.co.jp](mailto:shinmori_akihiro@intec-si.co.jp)

**Advait Siddharthan** University of Aberdeen, Aberdeen, UK, [Advait@abdn.ac.uk](mailto:Advait@abdn.ac.uk)

**Edlyn Simmons** Simmons Patent Information Service, LLC, Mason, OH, USA

**David Sinclair** Camtology Ltd, Cambridge, UK, [David.Sinclair@imense.co.uk](mailto:David.Sinclair@imense.co.uk)

**Mark Slater** University of Cambridge, Cambridge, UK, [Slater@hep.phy.cam.ac.uk](mailto:Slater@hep.phy.cam.ac.uk)

**Padmini Srinivasan** Computer Science Department and Informatics Program, The University of Iowa, Iowa City, IA, USA, [padmini-srinivasan@uiowa.edu](mailto:padmini-srinivasan@uiowa.edu)

**Veronika Stefanov** Information Retrieval Facility, Vienna, Austria, [v.stefanov@ir-facility.org](mailto:v.stefanov@ir-facility.org)

**Valentin Tablan** Department of Computer Science, University of Sheffield, Sheffield, UK, [V.Tablan@dcs.shef.ac.uk](mailto:V.Tablan@dcs.shef.ac.uk)

**John I. Tait** Information Retrieval Facility, Techgate, Donau City Strasse 1, Vienna, 1220, Austria, [john.tait@ir-facility.org](mailto:john.tait@ir-facility.org)

**Toshiyuki Takezawa** Hiroshima City University, 3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima 731-3194, Japan, [takezawa@hiroshima-cu.ac.jp](mailto:takezawa@hiroshima-cu.ac.jp)

**Hidekazu Tanigawa** IRD Patent Office, 8th floor, OMM Building, 1-7-31, Otemae, Chuo-ku, Osaka 540-0008, Japan, [htanigawa@ird-pat.com](mailto:htanigawa@ird-pat.com)

**Stephen Tomlinson** Open Text Corporation, Ottawa, Ontario, Canada, [stomlins@opentext.com](mailto:stomlins@opentext.com)

**Anthony Trippe** 3LP Advisors, Dublin, OH, USA, [tony@trippe.com](mailto:tony@trippe.com)

**Suzan Verberne** Computing Science Institute ICIS, Univ. of Nijmegen, Nijmegen, The Netherlands, [s.verberne@cs.ru.nl](mailto:s.verberne@cs.ru.nl)

**Merijn Vogel** Computing Science Institute ICIS, Univ. of Nijmegen, Nijmegen, The Netherlands, [merijnv@cs.ru.nl](mailto:merijnv@cs.ru.nl)

**Ellen M. Voorhees** NIST, Gaithersburg, MD 20879, USA, [Ellen.Voorhees@nist.gov](mailto:Ellen.Voorhees@nist.gov)

**Rebecca Watson** iLexIR Ltd, Cambridge, UK, [Bec.Watson@gmail.com](mailto:Bec.Watson@gmail.com)

**Peter Willett** Information School, University of Sheffield, 211 Portobello Street, Sheffield S1 4DP, UK

**Cynthia Barcelon Yang** Patent Information Users Group (PIUG), 505 Amberleigh Drive, Pennington, NJ, USA

**Veronika Zenz** max.recall information systems, Vienna, Austria, [v.zenz@max-recall.com](mailto:v.zenz@max-recall.com)

**Jianhan Zhu** True Knowledge Ltd., Cambridge, UK, [jianhanzhu@gmail.com](mailto:jianhanzhu@gmail.com)

**Part I**  
**Introduction to Patent Searching**

# Chapter 1

## Introduction to Patent Searching

### Practical Experience and Requirements for Searching the Patent Space

**Doreen Alberts, Cynthia Barcelon Yang, Denise Fobare-DePonio,  
Ken Koubek, Suzanne Robins, Matthew Rodgers, Edlyn Simmons,  
and Dominic DeMarco**

**Abstract** This chapter introduces patent searching in a way that should be accessible and useful to both researchers in information retrieval and other areas of computer science and professionals seeking to broaden their knowledge of patent searching. It gives an overview of the process of patent searching, including the different forms of patent searching. It goes on to describe the differences among different domains of patent search (engineering, chemicals, gene sequences and so on) and the tools currently used by searchers in the each domain. It concludes with an overview of open issues.

---

D. Alberts  
Theravance Inc., 901 Gateway Blvd., South San Francisco, CA, USA

C.B. Yang  
Patent Information Users Group (PIUG), 505 Amberleigh Drive, Pennington, NJ, USA

D. Fobare-DePonio  
Camarillo, CA, USA

K. Koubek  
Koubek Information Consulting Services LLC, Wilmington, DE, USA

S. Robins  
Patent Information Services, Inc., Westborough, MA, USA

M. Rodgers  
Landon IP, Alexandria, VA, USA

E. Simmons  
Simmons Patent Information Service, LLC, Mason, OH, USA

D. DeMarco  
DeMarco Intellectual Property, LLC, 1111 16th Street, South Arlington, VA, USA

M. Lupu et al. (eds.), *Current Challenges in Patent Information Retrieval*,  
The Information Retrieval Series 29,  
DOI [10.1007/978-3-642-19231-9\\_1](https://doi.org/10.1007/978-3-642-19231-9_1), © Springer-Verlag Berlin Heidelberg 2011



## 1.1 Introduction

Patents are legal documents issued by a government that grants a set of rights of exclusivity and protection to the owner of an invention. The right of exclusivity allows the patent owner to exclude others from making, using, selling, offering for sale, or importing the patented invention during the patent term, typically 20 years from the earliest filing date, and in the country or countries where patent protection exists. This temporary “monopoly” provides the patentee with a competitive advantage. Patent owners can also derive value from their inventions by licensing them to others who have the entrepreneurial capacity and innovative ability to develop, manufacture and market their inventions. In exchange for this right of exclusivity, the patentee is obligated to disclose to the public details of the invention, and related technical or scientific background information and state-of-the-art basis for the invention. Thus, patents typically contain more details and are more exhaustive than scientific papers. According to a United States Patent & Trademark Office (USPTO) study [1] published in the “Eighth Technology Assessment and Forecast Report”, patents provide a significant amount of unique and valuable technological information that is largely not available elsewhere.

First, it is important to consider the typical patent life cycle and become familiar with a few terms. Patents are granted by patenting authorities or central offices that are usually part of the national governments in hundreds of countries around the world. The process by which patenting authorities and inventors negotiate toward the terms of a patent is called *patent examination* and is also referred to as *patent prosecution*. Patent examiners, who are employed by a national or regional patenting authority, conduct patent examination. During examination, the patent examiner will search for prior art, or public disclosures of the features of the invention, that were available prior to the filing of the patent application. The examiner may also initially reject the patent application based on the similarity of the prior art uncovered during the search or provided by the inventor. An inventor may represent him- or herself to prosecute the patent application. Alternatively, the inventor may hire a patent attorney or patent agent, generically referred to as patent practitioners, to prosecute the application on the inventor’s behalf.

After a patent application undergoes examination and is deemed to satisfy the requirements for a patent set forth by the governing laws of the patenting authority, the patent may be used to enforce the right to exclude others from making, using, selling, and distributing the patented invention. After a patent is granted, the patent owner is usually required to pay maintenance fees to the granting patenting authority at certain intervals for the patent to remain enforceable. At this stage, and provided that all maintenance fees are paid, the patent is considered “in-force”, “active” or “live”. The patent may be asserted in a lawsuit against parties who are allegedly making, using, selling, or distributing the invention within the jurisdiction or country of the patenting authority that granted the patent, or the patent may be licensed for use by another party in exchange for a licensing fee. The patent may be enforced for the duration of its patent term—the limited period of time granted for the patent. Once the patent expires, the invention then belongs to the public or is “in the public

domain” and can be made, used, sold, or distributed by anyone. A company that is aggressive in enforcing their patents may frequently seek licensing agreements or file suit against others who are allegedly practicing the invention protected by their patents, and likewise may frequently engage in patent litigation.

There are many business and legal decisions that may need to be made throughout the patent life cycle. Even prior to having an invention, a company or individual may have the need to evaluate what has already been patented in their industry in order to know what areas of their industry to focus their innovating energy and resources. A company may already be involved in research and development for a technology or product and may need to know how they should design around the boundaries already protected by other in-force patents. When approaching a large product rollout, a company may need to conduct one last check to be sure that the features of the product can be made, used, sold, or distributed without infringing upon other in-force patents. The business decisions relating to product rollouts or product designs can have major financial implications. Prior to filing or even drafting a patent application, an inventor and their patent practitioner may want to gauge the success which the hypothetical patent application may have when it is sent to be examined by a patenting authority. In the preceding stages of either protecting a company’s patent portfolio or in seeking licensing agreements, the company may seek evidence of the company’s already patented technology being made, used, sold, or distributed by others. In the event that a company is sued by another for patent infringement the defendant may attempt to find prior art that precedes the plaintiff’s patents to demonstrate that the patents are invalid and unenforceable.

All of these business and legal needs bring us to the focus of this chapter and this book—they all require patent searching. While the term “patent searching” can mean “the act of searching patent information” or “searching for patents”, the phrase is more commonly used to describe *searching and filtering a body of information in light of and guided by an intellectual-property related determination*. This is the definition you should carry forward with you as you read this book. The business and legal needs above represent a variety of intellectual-property determinations, or drivers that render the need for patent searching.

The body of information invoked in our definition of patent searching can comprise any collection of published information, whether patents, peer-review papers, press releases, conference proceedings, industry standards definitions, product literature, product packaging, textbooks, drawings, diagrams, or anything that can adequately describe the subject matter at hand. The body of literature to be searched may change in scope and volume depending on the need for the patent search.

With more than one million patents applied for worldwide each year, the amount of information available to researchers and the opportunity to derive business value and market innovative new products from detailed inventions is huge. However, patent documents present several peculiarities and challenges to effective searching, analysis and management:

- They are written by patentees, who typically use their own lexicon in describing their inventive details.

- They often include different data types, typically drawings, mathematical formulas, biosequence listings, or chemical structures which require specific techniques for effective search and analysis.
- In addition to the standard metadata (e.g., title, abstract, publication date, applicants, inventors), patent offices typically assign some classification coding to assist in managing their examination workload and in searching patents, but these classification codes are not consistently applied or harmonized across different patenting offices.

This chapter describes the practical experiences in and requirements for effective searching, analysis, monitoring, and overall management of patent information, from the perspective of professional patent information users. It is not meant to be exhaustive, but rather to provide an overview of the key aspects and requirements for effective patent information search and analysis. The subject matter is subdivided into three general areas:

- Overview and requirements of different types and sources of information, types of searches, depending on the purpose of the retrieval, such as patentability or potential infringement.
- Description and requirements based on information management approaches, such as metadata or bibliographic data indexing, taxonomy, controlled vocabulary, value-added indexing and classification schemes.
- Considerations in and requirements for searching specialized invention technologies, such as chemical structures, biosequences, or device/engineering drawings.

The ultimate purpose is that this practical view along with the description of key requirements for effective retrieval of patent information would contribute toward advancement of emerging retrieval technologies to support the user in patent search, analysis and information management processes.

## 1.2 Information Types

For the purposes of patent searching and our discussions in this book, searchable information can be thought of in a few major buckets. Bear in mind that “searchable” more accurately means “accessible”, whether by actually searching an electronic database or by manually retrieving and reviewing technical journals in a library. We can group the basic buckets of searchable information by the extent to which each one is readily searchable (see Table 1.1). For convenience, we can call this “searchability”. The basic buckets are:

- Patent literature
- Technical journal-grade literature
- Everything else (press releases, conference proceedings, industry standards definitions, product literature, product packaging, text books, drawings, diagrams, etc.)

**Table 1.1** Searchability governed by the level of organization of the literature

Overall level of organization	Level of format uniformity	Accessibility	Level of consolidation	Searchability
<b>Patent literature</b>	High	High	High	High
<b>Technical journal-grade literature</b>				
Academic journal-grade literature/dissertations	Medium	Medium	Medium	Low
Industry journal-grade literature	Low	Low	Low	Low
<b>Everything else</b>				
Market information	Medium	Medium	Medium	Medium
Financial information	Medium	Medium	Medium	Medium
Legal	Medium	Medium	Medium	Medium
Press releases/news	Low	Medium	Medium	Low
Product literature/manuals	Low	Very Low	Very Low	Very Low

Patent literature refers to both granted patents and published patent applications. Both are available for searching at many of the world's patenting authorities. Technical journal-grade literature refers to organized papers written with a focus on a specific topic and usually published by a well-known periodic industry journal. Everything else refers to the catch-all bucket of any other type of disclosure of technical information that could exist. The types of searchable information have been broken down into these categories simply due to the distinct levels of organization that can be seen in each one.

The "searchability" of each bucket is governed by the level of organization of the literature in each bucket, the level of format uniformity between individual documents, the accessibility of the literature in each bucket, and how consolidated the various avenues to search the literature in each bucket have become.

Patent literature is one of the most highly concentrated collections of technical information available in the world. It enjoys a high level of organization due to the various patent classification systems used globally. In addition many patents are marked as being member of patent families linking patents for the same invention but accepted in different jurisdictions or countries.<sup>1</sup> The level of format uniformity between individual documents is extremely consistent compared to other types of literature. Even comparing two patent documents that originated from two different patenting authorities, the format and arrangement is highly similar between documents. For example patents always contain extensive bibliographic information, a title, and abstract, a set of claims specifying the claimed scope of the invention, and background information. This enables electronic patent data to be arranged in quite a number of discrete data fields that can be searched individually or strategically

<sup>1</sup>See: <http://www.epo.org/patents/patent-information/about/families.html> (Accessed 15 Dec 2010).

together. Patent data are both very accessible and consolidated since much of them is either freely available via portals provided by patenting authorities or by commercially available search engines that serve as “meta” search engines enabling the user to search globally through one interface. Commercial search engines have brought a high level of consolidation to patent data and much of them can be accessed using very few separate channels.

Technical journal-grade literature has benefited from some organization and some uniformity. Some very common value-add collections like Ei Compendex by Elsevier leverage classification and theme-based organizational schemes. The level of uniformity between documents is mostly consistent, however, the data fields that journal-grade literature documents have in common are many fewer than patent documents. This yields fewer and less sophisticated options to search the data. Journal-grade literature is graded as moderately accessible since, while a large amount of literature has been aggregated in collections like Compendex, a world of un-digitized and un-abstracted literature still exists in manual, paper collections. Journal-grade literature suffers significantly from the fact that literature aggregators like Elsevier and Dialog that supply journal title collections in “files” limit the transparency the user has in knowing what is actually being searched and what the overlap is between data files and collections from other providers. *A searcher’s efficiency drops significantly when the exact scope of the information being searched is unknown.* For this reason, the level of consolidation of journal-grade literature is low since an effective search requires a far greater number of unique access points than patent literature to be effective.

All other forms of literature are scattered across all reaches of resources and locations. Collections such as press releases and conference proceedings are consolidated individually, but under most circumstances need to be searched separately from all other sources. Product literature and product catalogs are perhaps the least searchable of all valuable literature resources.

### 1.3 Information Sources

What sources to search is dictated by what type of search is required, the legal and financial implications of the search, how much time to complete it and how much one is willing to pay to get the information needed. The sayings “You get what you pay for” and “Buyer beware” are important to keep in mind when choosing sources. Fee-based sources are not always complete just as free sources are not always erroneous and incomplete. It depends on the searcher’s comfort level. Searching both types of sources would give a sense of how complete the search is. But how does one know if he/she has done as thorough a search as one can? One criterion is when the same answers are retrieved from different sources, regardless of cost.

The following are issues patent searchers generally consider when reviewing whether to use services that are fee based, or services that are free at the point of use.

- Patenting authorities offer free searching; however, coverage is limited to the authority's specific country or jurisdiction only. When looking for legal and prosecution history, these sites are invaluable.
- Fee-based search services tend to cover multiple databases and are more comprehensive.
- Customizations, such as linking to other sources are available from fee-based services.
- Precision searching, as well as advanced search and analysis features tend to be available more often from fee-based sources.
- Fee-based sources tend to have reliable servers.
- Users of fee-based sources have input in the product updates and development with respect to timeliness, comprehensiveness and user interface.

Also note that sources differ in:

- Quality, comprehensiveness and types of content
- Time coverage
- Indexing
- Timeliness
- Ability to search a number of databases at the same time and remove duplicates to get unique answers
- Cost
- Post search analysis features

Finding relevant information has been compared to finding a needle in the haystack. No one can argue that there is not enough information out there. It is important to be able to search the whole document in addition to indexed fields, which is an issue in some services. Further the freshness and coverage of the data need to be considered.

## 1.4 Patent Search Types

This section discusses attributes of state-of-the-art, patentability, validity, freedom to operate, and due diligence searches. Common elements that need to be identified for all of these searches are: the purpose, time coverage, and the most relevant sources to search.

Before proceeding further, it is important to state upfront the basic assumptions and principles of the patent searching process: No search is 100% complete. For patentability type of searches (see Table 1.2), the goal is to conduct a better search than the Patent examiner. For other patent search types, the goal is to be as complete as the resources and time allow.

When conducting a patent search, three factors will affect the results: cost, quality, and time:

### Cost

- Fee-based sources vs. free sources

- Complexity of the search
- Technical expertise and proficiency of the searcher

### **Quality**

- Technical expertise and proficiency of the searcher (whether employed in-house or outsourced)
- Database content and integrity, indexing quality

### **Time**

- Searching is an iterative process; allocating enough time to discuss the search request with the requestor is important.
- Exhaustive search and analysis—the chance of missing a relevant publication is less for a 20 hour search vs. a 2-hour search.

A brief introduction to the major search types is worthwhile to understand generally when the major collections of information should be searched. Table 1.2 below summarizes the main search types, their purposes, and literature collections that are appropriate (but not always practical to search) for each one. As you gain exposure to the field, you will see that the names associated with some search types can be either interchangeable or distinctly different depending on whom you consult. For example: state-of-the-art searches and evidence-of-use searches are closely related, as are pre-filing-patentability and patentability-or-novelty searches. Sometimes these terms are used interchangeably. Also bear in mind that the table below is a summary. There are many caveats associated with the criteria of applicable information for each search type that depend upon the governing laws of each patenting authority. There are also arguably many additional search types. These are only the most common.

#### ***1.4.1 State of the Art Patent Search (Evidence of Use Search)***

The purpose of the State of the Art search is to gain comprehensive overview of a product or technology. Ideally this search is done before any R&D investment is made. In some companies, results from this search impact the selection and funding of a new project. This search is also useful when looking for a technology to license. This comprehensive search typically includes patent and non-patent literature sources. The interview process is critical in order to develop the appropriate search strategy, which tends to be broad. The data set retrieved can be large. The searcher needs to have a good understanding of what the requestor is looking for to enable quick review of the answers for relevancy. Another way to digest the result is to sort references using “patent” as document type. It is fairly easy to rank by assignees, inventors and patent classification codes. From the tabular list, one will be able to identify competitors, technology experts and technology fields. When the patent search results are analyzed using graphics and charts to visualize results, this

**Table 1.2** Types of patent searches

Search type	Purpose	Applicable information
State of the art search	To sample each major facet of a broad technology within a recent period. To gain comprehensive overview of a product or technology before any R&D investment is made or when looking for a technology to license.	All information published prior to today. Includes broad coverage of information and timeframe. Both patent and non-patent literature sources are included.
Evidence of use search	To identify literature supporting evidence that a product encompassed by the claims of an active subject patent is being made, used, sold, or distributed within the jurisdiction or country of origin of the patent.	Any literature published prior to the appropriate date associated with the subject patent.
Pre-filing patentability search	To identify prior art pertaining to both the core inventive concept and all sub-features for the purposes of drafting a patent application in light of the identified prior art.	Anything published prior to today.
Patentability or novelty search	To identify prior art pertaining to the core inventive concept of an invention that may preclude the invention from being patentable.	All information published prior to today.
Clearance or freedom to operate search	To identify any enforceable, granted patents claiming the subject matter of a product that is intended to be made, used, or sold, in a target jurisdiction or country.	Enforceable patents and published patent applications originating from only the target jurisdiction or country.
Validity or invalidity search	To identify prior art that describes the technology recited by the claims of a granted, target patent that would render the patent unpatentable as of the date it was applied for.	All information published prior to the appropriate date associated with the target patent.
Patent portfolio search, patent landscape search	The needs for landscape searches vary wildly and are typically business driven, to assess gaps of patent protection in an industry or comparing patent portfolios between two or more competitors.	Depends on purpose and extent of search: typically global patents and published patent applications and business data.



type of report is called a patent landscape analysis [2]. It is a graphical representation of how patent publications are related. There are a number of products [3] that specialize in patent landscape analysis, each with its own strengths and weaknesses. Using these tools, more elegant analysis is possible. For example, by looking at the level of patenting activity by classification codes over time one may get an insight on the maturity of the field as well as patenting trends identifying technology decay and rise. Just like any type of data analysis, the conclusion is only as good as the data set used. It is advisable to be cautious in drawing conclusions derived from patent landscape analysis. To be comprehensive, multiple sources should be searched. This introduces additional issues to consider in merging the results: (1) standardization of data fields to integrate the appropriate values from similar data fields; (2) duplicate removal and (3) one record per patent family representation to avoid skewing the analysis results.

### ***1.4.2 Patentability (Novelty)***

The purpose of a patentability search is to find all relevant prior art that may impact the likelihood of getting a patent granted. Issues such as novelty, non-obviousness and utility criteria need to be addressed. This type of search is typically conducted before writing the patent application, as the search results may change the scope of the claim or if needed lead to a ‘draft-around’. Since the coverage should include “everything made available to the public, in writing, by public use, or otherwise” [4], it is not enough just to rely on patent publications, books and refereed journal articles. Other atypical sources need to be searched as well: press releases, brochures, white paper, websites, conference materials, theses and dissertations, technical disclosures and defensive filings. For a typical patentability search, the searcher uses the following techniques:

- Keyword search
- Classification code (IPC, ECLA, F-terms) search
- Forward and backward citation of relevant documents
- Inventor or Author search of relevant documents
- Patent assignee search
- Chemical structure, sequence or mechanical drawing search, depending on nature of the request. Detailed descriptions of these specialized data types can be found in Sect. 1.7 and in Chap. 17 by Holliday and Willett in this book

### ***1.4.3 Freedom to Operate (Infringement, Right-to-Use, Clearance)***

The purpose of a freedom-to-operate search is to make sure that one does not infringe upon another’s patent that is still in-force. The focus of this search is on any granted patent that covers the invention and patent applications that may be granted

on the same invention. For patent applications, the search should include data from file wrapper and prosecution history. This type of search is country specific, so local agencies should be consulted to confirm the status of the patent. In addition, close attention to the patent claims is prudent since they may change from country to country. Although results may technically be limited to the last 20 years, it is wise to limit results to the last 25 years [5]. When conducting a freedom-to-operate search, the scope of the claim is the key. It is best not to limit the search to patents on the product itself, but also look at the processes needed to manufacture it, including everything from raw materials to packaging designs. For a typical freedom-to-operate search, the following attributes are also searched:

- Ownership/Patent assignee
- Patent family
- File history
- Legal status (e.g., patent term extension)
- Maintenance fee payments

#### ***1.4.4 Validity (Invalidity, Enforcement Readiness)***

The purpose of a validity search is to determine if a patent already granted for an invention is valid. It is also a measure of the strength of a patent. All sources mentioned in the patentability search (Sect. 1.4.2) are searched. However, the timeframe of the search can be limited to those results published before the filing date and a number of years after the filing date. As a rule of thumb, five years after the filing date is a good start, however, this is subjective, so it would be wise to consult legal counsel. The immediate availability of information can be troublesome when identifying publication dates. Publishers like ACS and Springer offer ASAP and Online First articles, respectively. These are edited submissions that are published online ahead of print. Another potential problem is tracking Internet page changes and the time stamp for any modification on the webpage. A bigger issue is if the webpage has been removed altogether.

For patent publications, the focus is on the validity of each claim and not necessarily the general purpose of the patent. Keep in mind that if the search is based on a patent application, the claims may change from the time the application is submitted to the time the patent is granted. If the search covers patents in more than one country, the claims may be different from country to country. When starting with a specific patent, in addition to the sources mentioned in the patentability search, the searcher also needs to consider:

- Non-Latin language publications
- File history (found in, e.g., US Patent Application Information Retrieval (PAIR), European Patent Office (EPO) Register and non-published patent office files)
- Examiner search history
- Documents cited by examiner and inventors
- Examiner's reason for allowance

### ***1.4.5 Patent Portfolio Analysis (Due Diligence, Patent Landscape Search)***

The purpose of a due diligence search is to assess if a company's patents are robust enough to exclude competitors and market the invention with the least probability of an infringement lawsuit. A due diligence search can be useful for companies looking to buy or partner with a company, and for companies who are looking to sell patents. A thorough due diligence search is expensive and will require a lot of time searching for and analyzing data. The question is not how much it will cost to do a due diligence search but "What is the cost of not doing a complete due diligence search?" A due diligence search is a Validity search plus freedom-to-operate search plus an analysis of the company's patent portfolio. The purpose of the deal and the results/findings from the due diligence search may guide investors in assigning a fair price to the desired product or technology. The buyer should not be the only party conducting a due diligence search. It is a good strategic move for the seller to conduct due diligence on its product or technology to ensure that the asking price is competitive.

## **1.5 Practical Considerations in the Searching Process**

No matter what type of search is requested, it is important for the patent searcher to really understand why the search is being requested. As mentioned earlier, searching is an iterative process [6]. Sometimes the requestor is not asking the correct question, so the interview process is critical. The searcher's knowledge of information resources that are available and past searches can be useful in defining the scope of the search. A searcher needs to be proficient in searching different information sources [7, 8] as well as possess technical or scientific background specific for the subject matter at hand. For example, having a degree in science is an advantage in pharmaceutical industry, but even with that, some level of specialization may be required. "You can teach a chemist how to conduct a structure search in less time than it takes to teach a non-chemist" [9].

The remainder of this introductory chapter will focus on the nuances and searching strategies associated with patent literature. As discussed, patent literature is highly organized, highly consolidated, and has very high consistency between documents. The major benefits that these characteristics bring to the "searchability" of patent literature are that a highly systematic methodology can be used. Searchers of patent literature have a number of valuable tools at their disposal: Citation Searching, Bibliographic Data Searching, Classification Searching, Full-Text Searching.

Before the influx of web-based applications and search tools, a searcher only needed to be proficient in using command lines to search STN, Dialog and Questel Orbit databases. This is not the case anymore. Stand-alone products are getting more and more popular and a searcher has to learn how to search each product well. Internet searching opens a whole new world of information. Occasional users and

accidental searchers prefer the Internet. When they are asked why, the most popular answer is “I always get an answer when I search the Internet”. The answer set might be full of false hits but they prefer that to getting no answer. But then how many of us have found an important document serendipitously on the Internet? Since it is free, the Internet should be searched first to gauge how much information is out there. Some results may be full-text documents, which may provide the searcher with better keywords to use.

Selection of search tools will depend on:

- Types of subject matter inventions
  - Chemical Structures
  - Biosequence data
  - Device/Mechanical/Electrical drawings
    - Section 1.7 describes in more detail these subject matter inventions, and requirements for searching these specialized invention technologies.
- Search techniques desired or most appropriate
  - Boolean logic
  - Natural Language Processing or Semantic technologies
  - Similarity
  - Proximity
  - Linking to full-text documents, external and internal depositories
  - Left and right word truncation
  - Case sensitivity when searching for acronyms
  - Keyword and synonym selection
  - Search term weighing
  - Search guidance on the fly
  - Controlled vocabulary or value-added indexing
  - Chemical structure based on textual description
  - Foreign words and characters such as Greek alphabet; and mathematical symbols
  - Search limits by sentence, section, etc.
  - Multi-language search query or translation to English from non-Latin languages (e.g. Japanese Chinese, Korean)
- Post search analysis features
  - Relevancy ranking
  - Sorting features
  - Subject relatedness
  - Citation Mapping (Citing and Cited)
  - Concept Mapping
- Alerting features by
  - Keywords
  - Structures (biologics and small molecules)
  - Legal status
  - Classification codes

## 1.6 Information Retrieval Approaches to Patent Searching

This section describes the various methods for patent information retrieval that have traditionally been employed to achieve high recall and precision: full-text searching, bibliographic data indexing, use of taxonomy/controlled vocabulary and classification schemes and value-added indexing.

### 1.6.1 Classification Searching

Classification codes [10, 11] are created and maintained by each patenting authority for the purposes of organizing patent and applications according to their technical application, structural features, intended use, or the resulting product produced by a process. The major classification systems in use worldwide include the International Patent Classification (IPC) system, the European Classification (ECLA) system, the United States patent classification (USPC) system, and the Japanese File Index and F-Term (FI/F-Term) classification system. Many other patenting authorities maintain their own classification systems, however, these four are the systems predominantly used when publishing and classifying patent data. The US and Japan are singled out because the patent examiners in these countries rely heavily on their own classification codes to classify patents. Examiners in these countries classify their patent documents in IPC and sometimes ECLA classification areas as a secondary measure and not as precise as their native classification areas. Due to the staggering volume of patent data produced by these countries, a global classification search is not complete without a search specifically within the US and Japanese classification systems in addition to IPC and ECLA.

#### 1.6.1.1 International Patent Classification (IPC)

The International Patent Classification (IPC) [12] system was established under the 1971 treaty, and has replaced national classifications or supplements them over the years since. The schedule of classes under the IPC is a true taxonomy, dividing all areas of technology into eight sections (A–H), the sections subdivided by notations for class, subclass, group, and subgroup. The classification system was originally updated at 5-year intervals, retaining the existing hierarchy. With the eighth edition of the IPC, a reclassification system was established so that all patents in a database use the same version. One of the IPC codes assigned to the athletic shoe in Fig. 1.1 has the following definition:

SECTION A—HUMAN NECESSITIES  
 A43 FOOTWEAR  
 A43B characteristic features of footwear; parts of footwear  
 A43B 13/00 Soles; Sole and heel units  
 A43B 13/14 • characterized by the constructive form



(12) **United States Patent** (10) **Patent No.:** **US 7,594,345 B2**  
**Fusco** (45) **Date of Patent:** **Sep. 29, 2009**

(54) **ARTICLE OF FOOTWEAR HAVING SOLE WITH RIBBED STRUCTURE**  
 (75) Inventor: **Ciro Fusco**, Portland, OR (US)  
 (73) Assignee: **NIKE, Inc.**, Beaverton, OR (US)  
 (\* ) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 200 days.  
 (21) Appl. No.: **11/247,591**  
 (22) Filed: **Oct. 12, 2005**  
 (65) **Prior Publication Data**  
 US 2007/0079530 A1 Apr. 12, 2007

4,546,559	A *	10/1985	Dassler	36/129
4,578,883	A	4/1986	Dassler	
4,615,126	A *	10/1986	Mathews	36/102
D292,443	S	10/1987	Ihlenburg	
D302,900	S	8/1989	Kolman et al.	
4,972,613	A *	11/1990	Loveder	36/105
D395,743	S *	7/1998	Ryan	D2/960
D399,342	S	10/1998	Carlson	
5,829,172	A *	11/1998	Kaneko	36/108
D405,597	S	2/1999	Carlson	
D471,347	S *	3/2003	Haas et al.	D2/953
D487,331	S *	3/2004	Rogers et al.	D2/952
6,793,996	B1	9/2004	Umezawa	
6,857,205	B1 *	2/2005	Fusco et al.	36/114
D512,553	S *	12/2005	Robbins et al.	D2/954
2003/0131499	A1 *	7/2003	Silverman	36/88
2004/0111922	A1 *	6/2004	Fusco	36/59 R
2005/0155254	A1 *	7/2005	Smith et al.	36/28

(51) **Int. Cl.**  
**A43C 15/02** (2006.01)  
**A43B 13/14** (2006.01)  
 (52) **U.S. Cl.** ..... **36/59 R; 36/59 C; 36/103**  
 (58) **Field of Classification Search** ..... **36/59 R, 36/59 C, 129, 25 R, 103; D2/953, 960**  
 See application file for complete search history.

\* cited by examiner  
*Primary Examiner*—Ted Kavanaugh  
 (74) *Attorney, Agent, or Firm*—Banner & Witcoff, Ltd

(56) **References Cited**  
 U.S. PATENT DOCUMENTS  
 D172,787 S 8/1954 Frary

(57) **ABSTRACT**  
 An article of athletic footwear comprising an upper for receiving the foot of a wearer and a sole structure attached to the upper, the sole structure having a heel portion including a rigid or semi-rigid ground contacting surface having a plurality of ribs located in the heel portion.  
**34 Claims, 3 Drawing Sheets**

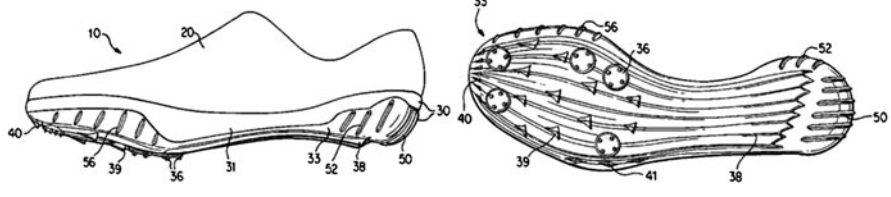


Fig. 1.1 First page of US 7,594,345 B2, assigned to Nike, Inc., published 29 September 2009

### 1.6.1.2 United States National Classification

The United States has continued to use its national system as the primary classification for patents [13]. The system consists of 3-digit class definitions, arranged numerically without any attempt to relate the numerical class code to its place in the sequence, creating new classes as new technologies emerge. Each class code is followed by a hierarchy of numerical subclasses, for example, class 36/129 for the athletic shoe in Fig. 1.1:

```

CLASS 36 BOOTS, SHOES, AND LEGGINS
83 BOOTS AND SHOES
113  · Occupational or athletic shoe (e.g., roof climbing, gardening, etc.)
114  .. Athletic shoe or attachment therefore
129  ... For track

```

### 1.6.1.3 European Patent Classification (ECLA)

The European Patent Office created a more precise variant of the IPC, assigning it to all of the patents in the Examiner search files [14]. ECLA codes do not appear on printed patents, but they are added to some databases. The ECLA code assigned to the athletic shoe in Fig. 1.1 has the definition shown below, a narrower definition than the IPC code shown above:

```

SECTION A—HUMAN NECESSITIES
A43          FOOTWEAR
A43B        characteristic features of footwear; parts of footwear
A43B 13/00  Soles; Sole and heel units
A43B 13/22  · soles made slip-preventing or wear-resisting, e.g., by
             impregnation or spreading a wear-resisting layer
A43B 13/22B .. Profiled soles
A43B 13/22B2 ... the profile being made in the foot facing surface

```

### 1.6.1.4 Japanese File Index Classification

FI terms are a system of refinements to the IPC, applied by the Japanese Patent Office (JPO) to Japanese patent documents [15]. The JPO also applies supplementary indexing terms, called F-terms, in addition to IPC and FI classifications, to assist in searching Japanese patent documents.

## 1.6.2 Full-Text Searching

Another significant benefit of patent data, in contrast to journal-grade literature is the wide availability of full document text among the major patenting authorities [16, 17]. Other forms of organized literature are often only abstracted. While for a number of years the bulk of full-text patent data were confined to the major seven patenting authorities (Europe, World Intellectual Property Organization, Germany, France, Great Britain, US, and Japan), many more patenting authorities are beginning to make their full-text patent data available. As well, several of the commercial data aggregators are translating the patent information from dozens of less conventional patenting authorities and making the data available within their search systems along side the major seven.

The text of patents differs in significant ways from the text of other forms of scholarly publications. The objective of patents is to obtain the patent owner's right to exclude others from practicing an invention described in the Claims section of the patent specification, and the patent laws and regulations of the country or patenting authority in which the patent application is filed largely control the language and formatting of the text. Customary phrases and sentence structure known as "patentese" are used in patent documents and are seldom used in other types of documents. There is no editorial process comparable to peer review before patent documents are published. The specification of a patent application is usually published 18 months after the first filing of an application covering the claimed invention, without any changes from the document filed by the applicant. The patent application undergoes examination to determine whether the claims define a patentable invention. Deficiencies in meeting the legal requirements for a patentable invention will prevent grant of patent rights and the publication of a granted patent, but pre-grant publication occurs whether the specification is well written or not.

The technical disclosure of a patent specification is provided in an Abstract, Claims, and the main body of the specification, which is often divided into sections:

- *The background of the invention:* a summary of the problem to be solved, ways it has been handled in the past and relevant prior publications.
- *A brief summary of the invention:* a short description of the invention being claimed, often a restatement of the Claims.
- *A detailed description of the invention:* a full description of all aspects of the invention, with definitions of the terms used and specific examples of ways in which the invention may be carried out. The description may be a few paragraphs or thousands of pages long. It may refer to defined terms and images in drawing pages or to chemical structures.

Because the patent Claims define the owner's right to exclude others from practicing an invention, patentees attempt to define their inventions in the broadest language possible. To expand the scope of a patent, the claims and accompanying disclosure often use generic language in place of simple terms. Shoes will be described as "footwear," house paint as "exterior finish," pills as "unitary dosage forms," and computer as "a system having a storage for storing data, an output device to display



information, a terminal for entering information, and a component that modifies the input data and controls flow of data between different parts”. Any terminology that defines the invention unambiguously is acceptable, and new technologies often require new terminology. Patent attorneys and agents often create new terminology to describe their clients’ inventions under the rule that “the patentee is his own lexicographer”. A full-text search must include any and all terms and phrases that may have been used to describe the technology of interest.

Patent claims are listed in the form of single sentences, with the leading phrase, “I claim,” implied or preceding the numbered list of claims. The precise wording and punctuation of the claims is essential to the understanding of the scope of legal protection, as is the meaning of each term defined in the specification. Claims may be “independent”, where all limitations of the claim are stated, or “dependent,” where limitations are carried over from an earlier claim. The entire text of independent claims is implied, but not stated in their dependent claims, so attempting to search the claim text using proximity operators often misses important references. An example of an independent claim and one of its dependent claims of the exemplary patent shown in Fig. 1.1 is shown below. Note that Claim 3 must be read as including the entire description given in Claim 1, with the added feature that the sole of the shoe is comprised of a polyamide. The word “comprising” is understood as meaning “including, but not limited to”.

I claim:

1. An article of athletic footwear comprising an upper for receiving the foot of a wearer and a sole structure attached to the upper, the sole structure having a heel portion, the sole structure including a rigid or semi-rigid ground-contacting surface, wherein a plurality of distinct ribs is located longitudinally in the heel portion and each of the distinct ribs extends in a substantially parallel direction, wherein the heel portion is cup-shaped so that the back portion of the heel portion extends upwards from a bottom portion of the ground-contacting surface and wraps around the backside of the heel, wherein at least a portion of the plurality of ribs curve around the back portion of the heel portion; wherein the plurality of ribs comprises a slippery material.

3. The article of footwear of claim 1 wherein the ground-contacting surface comprises a polyamide.

(Sample independent and dependent claim language: US 7,594,345 B2, assigned to Nike, Inc., published 29 September 2009. Article of footwear having sole with ribbed structure.)

Patent documents are written in the language of the patent issuing authority, and a multinational database will contain documents written in many languages and alphabets. In addition to countries that specify a single language, for example English in the United States and Japanese in Japan, there are some countries and international patenting authorities that allow the applicant to file a patent specification in one of several languages: the Patent Cooperation Treaty (PCT) allows applicants to

file applications in any of 10 languages as of 2010, including Japanese, Chinese, Korean, Russian and Arabic, as well as languages written in the Latin alphabet. Databases of PCT applications provide English-language abstracts, and many other databases also add English-language abstracts to the native language text records or substitute an English-language abstract for the patent text, but a search in English misses potentially relevant documents in other languages. The growing availability of machine translations helps to overcome the language barrier in databases that provide them, but the grammar and choice of words given by a machine translation engine often differ from those intended by the patentee.

Patents cover all technologies and even methods of doing business, and each area of technology has its own terminology in every language, often giving words a different meaning from their ordinary dictionary definition. The English word “furnish”, for example, is used in the papermaking industry to indicate the materials of which paper is made. Unless a search is limited to the technological context of the subject matter being searched, the results will not be sufficiently precise. Better precision can be achieved by searching text terms in combination with patent classification codes or other indications of context.

Searching full-text patent data requires a careful strategy and being constantly mindful of how a technology can be described from a scientist’s or engineer’s perspective versus how a technology can be described in the language of patent practitioners. The following key measures must be taken when leveraging the full-text data available in patent literature.

- Exhaustive usage of synonyms
- Effective use of Boolean operators, proximity operators, and truncation operators
- Appropriate clustering of concepts into discrete search queries
- Combining saved search queries appropriately
- Appropriate usage of broad-to-narrow and narrow-to-broad search query progression
- Iterative modification of previously stored search queries in light of newly acquired phrases and terminology

What are the pros and cons of full-text searching?

- Pros:
  - Easy to perform, no search training required
  - Allows for serendipity in searching
- Cons:
  - Optical character resolution (OCR) errors for those patents from countries/time ranges that are not created from original digital records
  - High recall, therefore relevancy ranking is needed
  - When searching for numbers—numeric versus text
  - Less precision: no control on which portion of the document the keyword appears as long as it is present in any part of the document

### ***1.6.3 Citation Searching***

Patents originating from the vast majority of patenting authorities are issued with a list of other documents that were cited during the prosecution of the patent application either by the patent examiner or the patent practitioner or inventor. Since the migration of patent information into electronic form, a patent searcher not only has immediate access to documents cited by patents but also immediate access to documents that cite each patent. The processes of searching both of these sets of documents are referred to as “backward citation searching” and “forward citation searching”. Backward referring to the documents a patent cites, and forward referring to the documents citing the patent under review. Citation searching is a patent searcher’s most powerful tool in quickly generating a highly concentrated collection of relevant search results at the beginning of a search. Search engine providers are making citation searching easier and easier. A common search strategy in beginning a search is to conduct a highly targeted search of only very relevant patent documents and then citation searching the most closely related documents for others of interest. Searchers can “follow their nose” through multiple generations of patent citations both forward and backward to rapidly collect highly relevant documents [18].

### ***1.6.4 Bibliographic Data Indexing & Searching***

The first page of a patent document includes bibliographic data relating to the filing details and ownership of the patent and includes additional data fields relating to the handling of the application within the patent office. References cited by the patent examiner may appear either on the cover page of the patent or in a search report appended to the patent publication. Databases index these metadata fields to facilitate searching. Bibliographic information is the focus of due diligence searching and some technical and competitive intelligence studies. Even in full text searches, combining keywords with bibliographic data, especially patent classification codes, can increase precision and limit search results to a desired range of filing or publication dates.

- **Title** Patent documents are required to have a descriptive title. Although patent regulations state that the title should reflect the claims, most original titles are relatively short and only hint at the novel features of the patent. Commercial databases may provide enhanced titles; in the case of the Derwent World Patents Index the title is an English-language mini-abstract of the patent.
- **Patentee (Applicant, Assignee)** The patentee is the owner of the patent rights, either the company or institution that sponsored the research leading to the patent or the individual inventor or inventors. Patent databases normally index the patent owner or assignee named on the patent document at the time of publication. Some databases apply standardized or normalized versions of the patentee name as an

aid to searching or apply company coding that attempts to track corporate divisions and ownership changes over time. Some databases supplement records with the name of organizations to which patent rights were reassigned after publication, obtaining the data from other patent office databases.

- **Inventor** The inventor or joint inventors are named on a patent document. Unlike the authors of journal articles, only individuals who contributed to the conception of the invention should be included.
- **Patent publication number** The serial number assigned by a patent office to the patent publication.
- **Publication date** The date on which the patent issuing authority published either the patent document or an announcement of the patent document in an official gazette. The publication date of most granted patents is the date when exclusive rights begin.
- **Application number** The serial number assigned to a patent application when it is filed at the patent office.
- **Application date** The date on which the patent application corresponding to the published document was filed at the patent office.
- **Designated states** Patent Cooperation Treaty applications and regional patenting authorities list the names or ISO country codes of the states for which the application of the patent is effective.
- **Priority applications** The Paris Convention for the Protection of Industrial Property, the World Trade Organization and other treaties allow patent applicants to file applications on a single invention in member countries within a year of a first patent filing by claiming priority based on the application filed in the first country. The application numbers of the applications claimed for priority are shown as priority application numbers in the records on the later patent applications.
- **Patent family members** Patents based on the same priority applications form a patent family of patent publications from multiple countries covering aspects of the same invention. Some databases combine all family members in a single record and apply indexing to a single patent document, known as “the basic patent.”
- **Priority dates** The filing dates of the applications claimed for priority.
- **Patent classification codes** The national and/or international classification codes assigned to the patent at the time of publication are printed on the patent specification at the time of publication. Some patent databases enhance the classification data by adding classification codes assigned by patent offices during post publication reclassification procedures.
- **Cited references** Patent examiners perform a search of the prior art to determine whether patent claims are new and inventive as defined by the patent law. Prior publications that teach or suggest aspects of the claimed invention are provided to the applicant for discussion and possible amendment of the application, and are listed on the patent document if it is eventually published. In addition to the cited references, some patent databases obtain information about later citations of the patent and add the citing patents to the record.
- **Additional search fields** Patent offices print the names of the patent applicant’s legal representative and the patent examiner on the patent document, and these

are included in the records of some, but not all, patent databases. Changes in the legal status of a patent or published application are included in some databases, in many cases obtaining the data from the European Patent Office's INPADOC legal status file.

#### **1.6.4.1 Important Preliminary Considerations of Searching Bibliographic Data**

Searching bibliographic data includes the ability to research prolific corporate entities and inventors who are known to have patented frequently in a given technology. However, searching for companies and inventors is not quite as simple as typing in the company or inventor name into a field. A number of precautions need to be taken into account when searching for specific names:

- Patent ownership can change frequently. A search for a company name may yield only older patents originally assigned to the company and not newer patents re-assigned to them after issuance.
- Company subsidiaries change frequently. Individual business units are bought and sold regularly. Further, searching only for a parent company name may not necessarily capture all company subsidiaries.
- Company suffixes (e.g., Co., Inc.) vary wildly and must be accounted for.
- Inventor names are commonly spelled in a wide variety of fashions, with and without suffixes, with or without initials, or completely misspelled altogether.
- Patents are very often not printed with assignment data upon issuance such that the owner files assignment data after printing.
- Correspondence address information can sometimes be used to approximate the ownership of patents.

What are the pros and cons of bibliometric data & abstract searching?

- Pros:
  - Errors in documents can be detected during database creation
  - Keyword synonyms and thesaurus available
  - Specific data fields like classification codes can be searched
- Cons:
  - Indexing errors can be introduced during database creation
  - Keywords that appear in non-indexed fields will not be searchable
  - Time lag from published date to database entry

#### **1.6.5 Taxonomy, Controlled Vocabulary and Other Value-Added Indexing**

In the days before full text searching was available, patent searchers were forced to rely on patent classification and controlled indexing systems for both manual and

online searches. The codes or terms were normally arranged hierarchically, permitting the searcher to use the narrowest appropriate term or the term at the latest position of a taxonomy, allowing the searcher to assemble a collection of documents, which would be reviewed for relevance, fully expecting that the limited number of indexing concepts would yield a great many irrelevant documents.

Patent classification systems were created for manual searching of printed patent collections. Patent offices designed numeric or alphanumeric schemes that assigned codes to all known technologies and marked each patent with one or more of the appropriate class codes. Class codes were updated periodically, creating a taxonomy by subdividing the classes to create collections of patents that were small enough that a searcher could review them. The schedules of class definitions formed a controlled vocabulary of generic terminology for each category of technology, and knowledgeable indexers and patent search specialists were able to select the nearest class definition for an invention of interest. Using the proper class code, one would be able to limit a search for a shoe sole such as the one in Fig. 1.1 without knowing whether the patentee called it a shoe sole, a ground-contacting surface, *une semelle* or *eine Schuhsohle*, and without retrieving patents on fishing gear.

Subject-based databases, such as Chemical Abstracts, and commercial patent databases, such as IFI CLAIMS, created systems of controlled indexing terms, applying the terms to indexing records in place of the actual terminology used by the patentee. The controlled indexing terms are collected in thesauri or vocabulary listings, which may be organized into a taxonomy in which broader, narrower or related terms are listed for each of the controlled indexing terms. A searcher can use the controlled terms from the thesaurus without having to create an exhaustive list of synonyms.

Chemical formulas lend themselves particularly well to controlled indexing, as there are a finite number of elements, and the empirical formulas of molecules disclosed in a document can be organized in a standardized alphanumeric fashion, for example the Hill system created for the Chemical Abstracts Formula Index. The molecular structures of chemical substances can also be indexed into systematic hierarchical systems, substituting a controlled indexing name or Registry Number for whatever name is used by the author or patentee in a document.

### 1.6.5.1 Chemical Substance Registries

A more precise system for retrieving information about chemical substances than a molecular formula or substance name is a registry system that gives a unique identifier to each indexed substance. An indexer reads a patent or other publication, recognizes each substance from a name or chemical structure drawing, and assigns an existing registry number or creates a new one, allowing searchers to find all references to that substance in the database by searching for the chemical structure or name in a registry database and then using the registry numbers to search in the corresponding bibliographic database.

The largest chemical substance registry is the Chemical Abstracts Service (CAS) Registry [19], which covers both patents and non-patent literature from around the

world. It assigns a registry number to each unique substance exemplified in a publication or claimed in a patent. The Derwent Chemistry Resource (DCR) covers compounds claimed or exemplified in international patents indexed in the Derwent World Patents Index (DWPI) [20]. The IFI CLAIMS Compound Vocabulary [21] has compounds mentioned in five or more United States patent publications. Because their indexing policies and database coverage differ, the number of compounds listed in the various registries and the number of patents associated with them are very different.

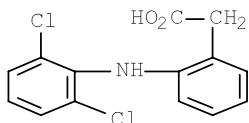
For example the non-steroidal anti-inflammatory drug diclofenac and its salts, has 299 registry numbers in the CAS Registry file, 24 registry numbers in the Derwent Chemistry Resource, and three registry numbers in the IFI Compound Vocabulary. Figure 1.3 illustrates the CAS Registry record for the acid form of diclofenac with its chemical structure diagram, a list of names that have appeared in the literature and the number of bibliographic records in the database indexed to this registry number. Searching the registry number 15307-86-5 in the Chemical Abstracts databases on STN (see Fig. 1.2) will retrieve all documents that disclose the acid form of diclofenac, regardless of the name used by the author of the original document, but it will not retrieve documents that disclose only the sodium salt of diclofenac, the active ingredient in Voltaren Gel.

### 1.6.5.2 Derwent Multipunch and Manual Codes

The Derwent World Patents Index was designed during the 1960's to facilitate in-house searching of English-language abstracts of chemical patents. The abstracts were printed on two types of card, IBM cards for sorting by use of a code represented by the positions of holes punched in the card, and Manual Code cards for searching by hand in file drawers.

The multipunch code was originally represented by 720 card positions, each position dedicated to a specific type of bibliographic data, chemical structure fragment, or other technical feature of an indexed patent. All of the codes relating to inventive features of the indexed patent were punched, and the searcher reviewed the abstracts of patents with all of the appropriate codes directly on the cards after they had passed through the sorter. After digital computers replaced card sorters the code was reformatted into alphanumeric symbols, and the code continues to be used. The chemical fragmentation section of the code is discussed in Sect. 1.7 below.

The Manual Code is a patent classification system, organizing technologies into a hierarchy that takes both structure and function into account. When it was used as a manual search tool, a searcher would identify a single code that best matched the inventive feature he or she wished to search and would visually scan through all of the abstracts in that section of the file drawer. Since the transition to computerized searching, Manual Codes have become a valuable tool for limiting retrieval in searches based on full text and keyword searches, specifying a required feature and eliminating all records covering features occurring higher in the hierarchy.



6291 REFERENCES IN FILE CA (1907 TO DATE)  
 212 REFERENCES TO NON-SPECIFIC DERIVATIVES IN FILE CA  
 6325 REFERENCES IN FILE CAPLUS (1907 TO DATE)

**Fig. 1.2** Chemical Abstracts Service Registry Database (CAS): structure record for diclofenac

L1 ANSWER 1 OF 1 REGISTRY COPYRIGHT 2010 ACS on STN  
 RN 15307-86-5 REGISTRY  
 CN Benzeneacetic acid, 2-[(2,6-dichlorophenyl)amino]- (CA INDEX NAME)

OTHER CA INDEX NAMES:  
 CN Acetic acid, [o-(2,6-dichloroanilino)phenyl]- (8CI)

OTHER NAMES:  
 CN 2-(2,6-Dichloroanilino)phenylacetic acid  
 CN 2-(2,6-Dichlorophenylamino)phenylacetic acid  
 CN 2-[(2,6-Dichlorophenyl)amino]benzeneacetic acid  
 CN 2-[2-(2,6-Dichlorophenylamino)phenyl]acetic acid  
 CN Dichlofenac  
 CN Diclac  
 CN Diclofenac  
 CN Diclofenac acid  
 CN Diclofenamic acid  
 CN Diclomelan  
 CN Dicloreuma  
 CN N-(2,6-Dichlorophenyl)-o-aminophenylacetic acid  
 CN Pennsaid  
 CN Transfenac  
 CN Voltaflan  
 CN [o-(2,6-Dichloroanilino)phenyl]acetic acid  
 DR 76595-40-9, 87180-41-4  
 MF C14 H11 Cl2 N O2

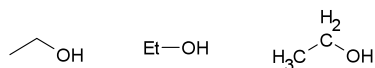
**Fig. 1.3** 15307-86-5 in the Chemical Abstracts databases on STN

## 1.7 Specialized Invention Technologies: Considerations & Requirements

While keywords and text terms are commonly employed in searching patents, certain subject matter inventions claimed in patents warrant specialized techniques for precise and high recall retrieval of relevant art. These include:



**Fig. 1.4** Different chemical structural representations of ethanol



- Chemical structures
- Biosequences and biotechnology topics
- Device/Engineering drawings

This section describes considerations and requirements for effective retrieval of these specialized invention technologies.

### 1.7.1 Chemical Structure Searching

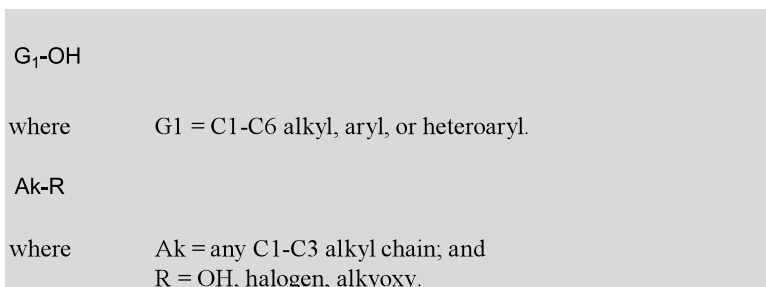
Searching for chemical compounds poses many challenges. There is wide variability in nomenclature, the search may be directed to a species or a genus that encompasses many possible species, and the chemical compound(s) of interest may be disclosed in a Markush structure.

Even exact compounds can be difficult to search and a professional searcher does not rely on chemical nomenclature for comprehensive retrieval. For example, something as simple as ethanol can be described as: ethanol, ethyl alcohol, grain alcohol, pure alcohol, hydroxyethane, drinking alcohol, ethyl hydrate and absolute alcohol.

Ethanol could be also depicted structurally, instead of mentioned by name, as shown in Fig. 1.4.

Exact compounds can also be described generically. For example, ethanol is a “hydroxy alkane”. Structurally, ethanol is also one of the compounds encompassed by either of the following two generics, as depicted in Fig. 1.5.

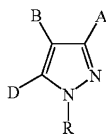
A search request may be a generic query, as shown in Fig. 1.5, that defines many possible compounds, in which case the goal is to retrieve records that relate to any of the compounds defined by the genus. This is called a Markush search [22]. The term “Markush” originated from the generic claims filed by Dr. Eugene A. Markush, which was granted as US 1,506,316 in 1924. A. Markush is essentially a way to claim many compounds in a single patent claim and the term is used to describe



**Fig. 1.5** Examples of genus representations of ethanol

The invention claimed is:

1. A Pyrazole derivative of formula (I), having affinity for the cannabinoidergic CB1 and/or CB2 receptors:



(I)

R is

aryl, not substituted or having from one to five substituents, equal to or different from each other, selected from halogen, C<sub>1</sub>-C<sub>7</sub> alkyl, C<sub>1</sub>-C<sub>7</sub> alkylthio, C<sub>1</sub>-C<sub>7</sub> alkoxy, C<sub>1</sub>-C<sub>7</sub> haloalkyl, C<sub>1</sub>-C<sub>7</sub> haloalkoxy, cyano, nitro, amino, N-alkylamino, N,N-dialkylamino, saturated or unsaturated heterocycle, and phenyl;

A is

an amidic substituent of formula —C(O)—NH-T', wherein T' is

a group NR<sub>1</sub>R<sub>2</sub>, wherein R<sub>1</sub> and R<sub>2</sub> are equal or different and have the following meanings:

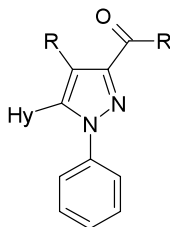
C<sub>1</sub>-C<sub>7</sub> alkyl; aryl, arylalkyl or arylalkenyl not substituted or optionally having on the aromatic rings from one to four substituents, equal to or different from each other, selected from halogen, C<sub>1</sub>-C<sub>7</sub> alkyl, C<sub>1</sub>-C<sub>7</sub> haloalkyl, C<sub>1</sub>-C<sub>7</sub> haloalkoxy, C<sub>1</sub>-C<sub>7</sub> alkylthio, C<sub>1</sub>-C<sub>7</sub> alkoxy, wherein in the previous substituents comprising C<sub>1</sub>-C<sub>7</sub> aliphatic chains, C<sub>1</sub>-C<sub>3</sub> chains are preferably used; wherein R<sub>1</sub> may additionally be hydrogen;

or R<sub>1</sub> and R<sub>2</sub> together with the nitrogen atom to which they are linked form a, saturated or unsaturated, heterocycle from 5 to 10 atoms comprising carbon atoms and including the nitrogen of NR<sub>1</sub>R<sub>2</sub>, and optionally an additional S, O or N atom, not substituted or optionally having from one to four substituents, equal to or different from each other, selected from C<sub>1</sub>-C<sub>7</sub> alkyl, phenyl, and benzyl, said phenyl or benzyl optionally substituted with one or more groups, equal to or different from each other, selected from: halogen, C<sub>1</sub>-C<sub>7</sub> alkyl, C<sub>1</sub>-C<sub>7</sub> haloalkyl, C<sub>1</sub>-C<sub>7</sub> haloalkoxy, C<sub>1</sub>-C<sub>7</sub> alkylthio and C<sub>1</sub>-C<sub>7</sub> alkoxy;

B is a group selected from: hydrogen and C<sub>1</sub>-C<sub>4</sub> alkyl; and D is an heteroaryl with a ring size of from 5 to 6 atoms, selected from the group consisting of thiophene, pyridine, furan, oxazole, thiazole, imidazole, pyrazole, isoxazole, isothiazole, triazole, pyridazine, pyrimidine, pyrazine, triazine and pyrrole; wherein the heteroaryl is optionally substituted with one, two, three or four substituents, equal to or different from each other, selected from the following: halogen, C<sub>1</sub>-C<sub>3</sub> alkyl, C<sub>1</sub>-C<sub>3</sub> alkylthio, C<sub>1</sub>-C<sub>3</sub> alkoxy, C<sub>1</sub>-C<sub>3</sub> haloalkyl, and C<sub>1</sub>-C<sub>3</sub> haloalkoxy.

**Fig. 1.6** An example of a Markush Claim from US 7,659,407

**Fig. 1.7** A typical chemical structure query



R = anything

Hy = any heteroaryl

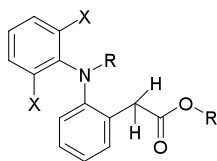
Leave all positions open to possible substitution but do not allow for ring fusion.

any generic structure that encompasses multiple species. An example of a Markush claim is shown in Fig. 1.6.

A typical patentability search request might be “find patents, patent applications, and literature references that claim or disclose compounds defined by the following generic” (as shown in Fig. 1.7), in which case the patent (US 7,659,407) above should be retrieved with the correct search query and appropriately indexed retrieval system.

Chemical structures represent molecules composed of atoms linked together by chemical bonds. There are groupings that occur in many molecules—rings of atoms, patterns of atoms and bonds that chemists refer to as “functional groups.” Database producers created indexing systems that fragment the molecules into their component rings and functional groups and assign an alphanumeric code to each of the resulting substructures. Indexers evaluate each chemical structure in a patent, and add all of the applicable codes to the database record. Some systems have been able to

**Fig. 1.8** Generic chemical structure search query for Diclofenac



X = any halogen

R = anything

The two phenyl rings may be further substituted but not fused.

partially automate this process. These fragmentation codes allow a searcher to look for either specific molecules or Markush structures with alternative substructures, using Boolean logic rather than resource-intensive structure searching algorithms. Because Markush structures often contain a great many alternative fragments, the systems include codes for fragments that are either required or optional in embodiments of the structure and a set of negation codes for fragments that can never be present in an embodiment.

There are different types of systems [23] available for searching chemical structures in the patent and non-patent published literature. Topological search systems are used to match graphical structures created by a searcher with specific compounds or Markush structures contained in a database. An indexer adds chemical structure indexing to the search system based on the indexer's understanding of the patent or literature document. Special software is used by the searcher to create the structure query. Chemical fragmentation code search systems, such as Derwent fragmentation codes and IFI Claims chemical vocabulary codes, match alphanumeric codes from strategies created by a searcher with codes added to a database record by indexers.

As computing systems advanced, connection tables were created that index how these atoms and groups of atoms are interlinked together to allow for more precise retrieval. For example, graphical searches in several structure searchable databases hosted by STN can be an Exact Search (EXA), which is used to retrieve substances that exactly match the query, a Family Search (FAM), which is used to retrieve substances that exactly match the query plus multicomponent substances such as salts, a Closed Substructure Search (CSS), which will retrieve substances that match the query without substitution allowed, or a Substructure Search (SSS), which will retrieve substances that match the query with any substitution allowed. To conduct a search using a structure query a searcher first creates the chemical structure query using software, such as STN Express.

### 1.7.1.1 Diclofenac Chemical Structure Search Strategy

An example is described here for conducting a chemical structure-based search of Diclofenac as a gel formulation in a freedom-to-operate assessment. There are two concepts to consider, the compound Diclofenac and gel formulation. Figure 1.8 exemplifies the chemical structure search strategy on how one might conduct a

freedom-to-operate search for Diclofenac. The second concept, gel formulations, could be searched using full-text searching, classification schemes, and other value-added indexing described earlier in Sect. 1.6.

The compound Diclofenac (2-[2-[(2,6-dichlorophenyl)amino]phenyl]acetic acid) is marketed under several tradenames, such as Voltaren and Caltaflam. Tradenames, chemical names, and synonyms would need to be identified and incorporated into the search. There are many ways to identify these names, such as reading the compound records found in Chemical Abstracts Registry File, Derwent World Patent Index, IFI Claims, Medline, Embase, and other free Internet sources. The compound registry numbers of Diclofenac applied by the database indexers would also be searched. An initial keyword search for Diclofenac in various databases could help identify some of the value-added indexing and classification available.

Searching for the exact compound alone is not sufficient for a freedom-to-operate search since the search must also retrieve patents with broad claims that encompass Diclofenac, so a generic search query is needed. An example of a generic query that encompasses Diclofenac is shown below in Fig. 1.8.

This generic query can be executed using the STN International system as depicted in Fig. 1.9.

This query could be searched in any of the Structure Searchable databases [24] hosted on STN such as Registry, Derwent DCR, Beilstein, and Marpat. Care must be taken when creating a search query since designations of bond types, match level, element count and connectivity can greatly alter the results. The above query searched as a substructure search (SSS) on STN would allow for substitution everywhere except at node 16, require the two rings to be isolated, and allow for retrieval of records in the Marpat database with broad claim language such as “aryl” for the phenyl rings and “electron withdrawing group” for the halogens.

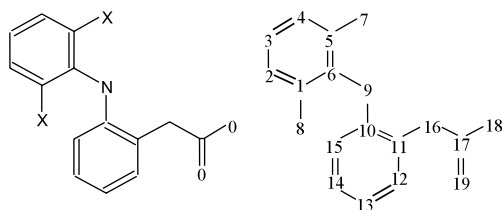
A searchable query for the above genus can also be executed using the Markush DARC system, as shown in Fig. 1.10.

The above query searched in Questel’s Merged Markush System (MMS) [25] would allow for substitution everywhere except at node 14, require the two rings to be isolated, and allow for retrieval of records that relate to compounds defined by the above generic either specifically or generically.

Chemical fragmentation code strategies should encompass the specific compound, as well as a generic representation. Figure 1.11 outlines a Derwent chemical fragmentation code strategy for Diclofenac to be searched in the World Patent Index (DWPI) database [26]. Figure 1.12 illustrates the IFICDB [27] chemical fragmentation code search for Diclofenac. The list of negation codes has been shortened due to space limitation.

Results from each of the above structure and chemical fragmentation code searches would be combined with the strategy for gel formulations, limited to patents or published patent applications, limited by country as requested, and then limited by date to capture patents that are still in force.

The above sample search strategy is not meant to be exhaustive, but rather to illustrate some of the common approaches taken when conducting a freedom-to-operate search that includes a chemical compound. Each type of search and each



chain nodes :

7 8 9 16 17 18 19

ring nodes :

1 2 3 4 5 6 10 11 12 13 14 15

chain bonds :

1-8 5-7 6-9 9-10 11-16 16-17 17-18 17-19

ring bonds :

1-2 1-6 2-3 3-4 4-5 5-6 10-11 10-15 11-12 12-13 13-14 14-15

exact/norm bonds :

6-9 9-10 17-18 17-19

exact bonds :

1-8 5-7 11-16 16-17

normalized bonds :

1-2 1-6 2-3 3-4 4-5 5-6 10-11 10-15 11-12 12-13 13-14 14-15

isolated ring systems :

containing 1 : 10 :

Connectivity :

16:2 E exact RC ring/chain

Match level :

1:CLASS 2:CLASS 3:CLASS 4:CLASS 5:CLASS 6:CLASS 7:Any 8:Any 9:CLASS 10:CLASS

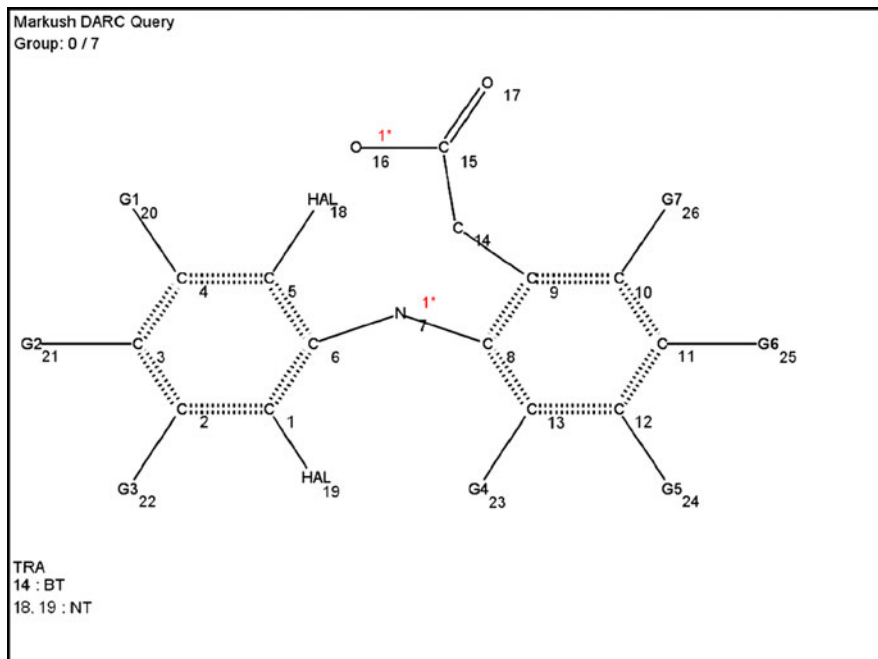
11:CLASS 12:CLASS 13:CLASS 14:CLASS 15:CLASS 16:CLASS 17:CLASS 18:CLASS

19:CLASS

**Fig. 1.9** STN Express generic (genus) structure query for Diclofenac

type of search system provides value. It is up to the searcher to determine, which type of searches, and which search systems to use for any given search. Whenever possible multiple databases and systems should be used since each system provides unique features, different coverage, and different indexing policies. It is also not uncommon for a chemical search to retrieve a wide range of records depending on the databases and systems used. For example, 50 patent family and literature records might be retrieved from searching a structure query in various databases while 1000 or more patent family records might be retrieved using fragmentation codes. The professional searcher must understand the details of how each of the systems works in order to explain and analyze these results properly.

In spite of recent advances in chemical structure searching there are still many areas that could be improved. For example, it would be useful for analysis purposes to be able to search and retrieve records with compounds of interest that are specifically claimed versus compounds that are only disclosed in the specifications, or to search and retrieve records with compounds of interest that are only claimed generically. It should be noted also that the indexing conducted by database producers is applied to the basic member of a family and not to each subsequent family mem-



G1-G7:

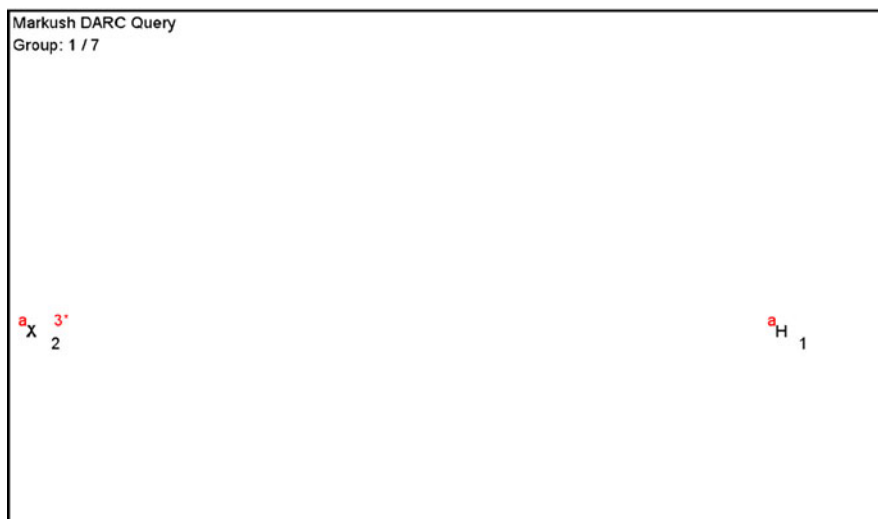


Fig. 1.10 Questel's MMS Markush DARC structure query for Diclofenac

ber added to a database record. Claim coverage can change from one document to another so it would be helpful to have every family member indexed. And finally, chemical concentrations or percentages are currently not indexed and often the nov-

```

=>S (G100(P)H141(P)H602(P)H608(P)J171(P)M414(P)M532)/M0,M2,M3 \>_line1
=>S _line1(P)(M121(P)M143)/M2,M3 \>_line2
=>S _line2(P)(M280(P)M311(P)M321(P)M342(P)M391(P)(M370 OR M372))/M2,M3 \>_line3
=>S _line3(P)(G011(P)G014(P)H102(P)H642(P)J011)/M2,M3 \>_line4
=>S (_line1(P)M900/M0) OR (_line2(P)M901/M2,M3) OR (_line3(P)M902/M2,M3) \>_line5
=>S _line5 OR _line4 \>_line6
=>S _line6(NOTP)(H2 OR H3 OR H4 OR H5 OR H7 OR H8 OR H9 OR J2 OR J3 OR J4)/M2,M3
\>_line7
=>S _line7(NOTP)(J5 OR J6 OR J9 OR K0)/M2,M3 \>_line8

```

**Fig. 1.11** Derwent chemical fragmentation code strategy in WPI for Diclofenac

```

S 30035/FG (L) 30047/FG (L) 30295/FG (L) 32742/FG (L) 34194/FG (L) 34701/FG (L) (10 or 20 or
30)/RL
S L1 (NOTL) 30037/FG (NOTL) 30040/FG (NOTL) 30039/FG (NOTL) 30038/FG
S L2 (NOTL) 34205/FG (NOTL) 30027/FG (NOTL) 34246/FG (NOTL) 31080/FG
[ 26 more lines of negation codes]

```

**Fig. 1.12** IFICDB chemical fragmentation code strategy for Diclofenac

elty of an invention is not a particular compound but rather its concentration in a formulation.

In conclusion, some minimum requirements for an effective chemical structure retrieval system are nomenclature searching that includes generic descriptions, the ability to search by chemical structure, and Markush searching. To be effective the database must also provide details of its indexing policies and any changes over time.

## 1.7.2 Biosequences/Biotechnology Searching

As with other domain searches for patent and scientific literature, a professional patent searcher in biotechnology must be able to perform comprehensive text word searches, utilize controlled vocabulary terminology, classification schemes, sequence code match techniques and algorithms for finding biosequence homology (similarity attributed to descent from a common ancestor [28]).

One of the difficulties for a biotechnology patent searcher is locating and compiling comprehensive data from many sources. Such information can be provided in different (non-) textual formats (articles, biological sequences, patent documents, tables summarizing and comparing biological data, images of biological samples, graphics representing experiments, etc.) and scattered among many types of publications and databases or published directly through the Internet [29].

### 1.7.2.1 Nomenclature Challenge

Similar to chemical substance nomenclature, locating gene or protein name is a challenge due to various nomenclature systems, aliases and sources needed to be

consulted. Genes can have several names, synonyms and redundant gene symbols. As an example, the human gene GBJ2 has several names and aliases/synonyms:

Gene Symbol: GBJ2  
Gene Name: gap junction protein, beta 2, 26kDa  
Previous gene symbols: DFNB1, DFNA3  
Previous gene names: gap junction protein, beta 2, 26kD; connexin 26, gap junction protein, beta 2, 26kDa  
Gene aliases: CX26, NSRD1

Professional searchers must give consideration if they need to include genetic alleles (phenotypic gene variation for example, green vs. blue eyes), if the request is for a specific species' gene (mouse vs. human gene) and mutated gene names. The names of protein and peptides have similar nomenclature issues. Protein receptors and their ligands can have similar names that can result in false hit retrieval. Recombinant proteins will also have different names and designated abbreviations. Determining a comprehensive search hedge (a collection of search terms) of nucleic or protein names is important for intellectual-property searches necessary to compliment a comprehensive biosequence search.

### 1.7.2.2 Biosequence Searching Considerations

Patent sequence information found in both commercial and public databases is not comprehensive [30]. A sequence of interest may or may not be disclosed in patent documents, necessitating the need for additional text word searches in combination with a biosequence search. Database inclusion of the sequences from a patent document is determined by the producer's indexing policies. A professional searcher will need to be aware of each system's indexing policies and limitations:

- Does the search system have a biosequence length limitation? Nucleic sequences are often long.
- Are all the sequences found in a patent publication indexed or are only the claimed sequences included in the database?
- What year did the publisher start including biosequences in their database?
- How are short biosequences indexed? Are the short sequences (<9 nucleic or amino acid units) included in the database as a sequence or is it necessary to search the biomolecule as a chemical structure?
- Mega sequence (containing many different sequences or a single extremely long sequence) patent documents may or may not be indexed in databases.

Biosequences are searched as either Sequence Code Match (SCM) or as a homology search. In code match searches, the search system aligns the query search sequence codes against a database of sequences by the code of each nucleic/amino acid unit. For example:





are based on % identity and/or length of certain amino acid regions [34]. GenomeQuest's GenePAST/Fragment search is based on GenomeQuest proprietary algorithm, which is defined as "The GenePAST percent identity" that finds the best fit between the query sequence and the subject sequence, and expresses the alignment as an exact percentage [35].

### 1.7.2.3 GBJ2 Biosequence Search Strategy

An example is described here on a sequence freedom-to-operate assessment based upon the genetic sequence of gap juncture protein beta 2 (GBJ2) and the protein for therapeutic use. GBJ2 may have a genetic component in hereditary deafness. The search can be accomplished by utilizing nucleotide sequence and amino acid homologous sequence searches. The professional searcher should consider additional search types, such as full-text, classification and value-added indexing searches for comprehension. There are publically available gene and protein database to assist the searcher in locating the gene and protein biosequences, names and synonyms, if necessary. Databases on National Center of Biotechnology Information, European Bioinformatics Institute, DNA Data Bank of Japan, The Jackson Laboratory, and other web-based sites are helpful in locating data and information on genes and proteins for search preparation.

Execution of biosequence homology searches would ideally be completed on publically available web sites and commercial databases. However, in many industrial companies, transmitting sequence data over the Internet is prohibited, so commercial databases are searched. Prior to the search, the professional will need to determine the relevant percent identity of similarity and the sequence length that is appropriate. BLAST and FASTA algorithms were designed for biological researchers and their needs, not for patent searchers and should be considered when analyzing the retrieval. If the biosequence is less than 30 residues in length, BLAST options need to be adjusted to retrieve the best hits, along with other sequence search strategies.

Homology sequence search is not comprehensive for a freedom-to-operate request. The search may need to cover genetic variants, chemically modified sequences, mutations and claims that discuss similar biological function but without a disclosed sequence. As with other patent searching, the biosequence search should include keyword or text-based, enhanced indexing, full-text and classification strategies. The above strategy is not meant to represent a comprehensive search but rather to illustrate factors to consider in constructing a search strategy.

Biosequence searching has improved in comprehensiveness of available data. However, there is a demand to include more sequence data from the whole patent document. It is not unusual to find the sequence of interest in a patent diagram. Comprehensive sequence data from every patent family member, not just the basic patent, are desirable from all database producers. USGENE includes sequence information from all the US patent family members and has increased patent biosequence data availability generated from the US Patent Office. Finally searchers and

patent analysts require additional similarity algorithms, which have the capability to search biosequences and deliver similarity scores in alignment with how claims are crafted.

In conclusion, biosequence databases contain incomplete information, and necessitates searching biosequences found in patent and scientific literature in both commercial and, if allowed, publicly available sources and systems. Complementing biosequence search with text-based search strategies is important for comprehensive retrieval and intellectual-property analysis. In addition, a professional searcher needs to have an understanding and working knowledge of the indexing policies and limitations of each database.

### *1.7.3 Searching Device/Engineering Drawings*

The retrieval of patent information within the disciplines of engineering, and more specifically the mechanical/electrical fields of engineering, is a case study in the application of the “No Free Lunch” theory [36]. The application of this premise to patent information retrieval is very clearly visible in the methodologies a professional patent searcher uses in locating art of relevance. From an initial keyword-based search limited to abstracts to full classification searches to multi-generational citation analysis to combining focused keywords with classification ranges, all of the algorithms used by a searcher are performed to search and retrieve more efficiently.

What is most overwhelming to an outsider used to the relative ease of locating information based upon words is the volume of references that are traditionally reviewed by a patent searcher within the engineering disciplines. For a single, relatively simple project, a mechanical patent searcher may manually review upwards of 5,000 patent documents to locate a mere 10 of particular relevance (manual review denotes a physical eyeballing of all figures, not a title-based review of the document). And why is this? Because traditional search engines and the algorithms employed are very inefficient in the engineering arts, which are heavily dependent upon drawings to clearly convey a concept [37].

While a physical picture may clearly show a car bumper, the text of a patent may describe a safety device for the protection of people or objects, said safety device utilizing multiple materials, said multiple materials comprising a rigid material and one or more less rigid materials, said rigid material selected from plastics and foam, said less rigid materials selected from plastics and foam. The picture may be recognized by a patent searcher in less than 3/10th of a second as external to motor vehicle, while the text could be parsed dozens of times for some hint as to whether they intended a bumper, an internal padded vehicle component, a helmet, or even shin guards for a soccer player.

The single biggest issue that causes this inefficiency and must always be noted with regard to search and retrieval of engineering drawings is that while “a picture may speak a thousand words”, it also does so in such a direct and succinct manner. In contrast, the mundane and simple can easily be transformed into the obtuse and unclear by a quality wordsmith or lexicographer (typically the patent attorney/agent).

Therefore, to avoid the dependency upon words, a searcher of patent information in the mechanical/electrical engineering disciplines learns to rely upon additional tools or algorithms for the location of relevant documents combined with rapid vetting via image analysis. These other algorithms are classification schemes described in Sect. 1.6.1, classification limited by keywords, and citation analysis. Then, using a circular flow path to emphasize the iterative process of searching, multiple iterations will be performed to locate the documents of relevance.

An additional important means for removing the dependency upon words within the mechanical and electrical engineering disciplines is the formulation of specific search strategies. Most inventions or improvements lend themselves to a formulaic combination of features: (A) Specific field of technology and (B) Problem to be solved and (C) Solution to be applied.

An ideal reference will encompass A, B, and C. Of almost equal relevance will be the subcombinations of A, B and C (A and B, A and C, B and C). This is particularly true when setting up a search strategy or field of search. For example, when C (solution to be applied) is best represented by a picture or figure, a search strategy must be set up to search for all documents with A (technology field) and B (problem to be solved). Those with C will inherently be included and only by manual review will C be recognized and identified.

Going further down the thought pattern, often multiple features (B and C for example) are poorly defined by anything other than a picture. Then a professional searcher must manually review all references within A (the field of technology) and examine the figures to identify those of interest to B and/or C. This is also the ideal time to apply our first mentioned means (Classification, Classification limited by keywords, and Citations) to avoid the dependency upon words and these are further detailed below:

**How Is Classification Used** Using the example above regarding a car bumper, the US classification schedule has a class (293) labeled “Vehicle Fenders”. With this Class 293, a range of subclasses in an outline format running from 102 to 155 is labeled “Buffer or Bumper Type”. What this means is that all patents classified in Class 293/Subclass 102 to Class 293/Subclass 155 are primarily focused upon Vehicle Fenders and specifically on Buffer or Bumpers (approximately 7,000 documents). And more specifically, subclass 120 depends from the broad subclass 102 and is titled “Composite bumper” which very closely reads upon the plastic and foam combination of rigid and less rigid materials. Thus, a review of the documents in Class 293/Subclass 120 will put over 700 documents of high relevance in front of a professional patent searcher without using a single keyword limitation.

**How Is Classification Limited by Keywords Used** Again using the bumper sample, the International Patent Classification (IPC) for Bumpers, which corresponds to the above-mentioned US Classification Range 293/102-155 is B60R19/02-19/50. More specifically, subclasses 19/03 reads on composite bumpers, 19/18 reads on impact absorbing, and 19/22 expressly reads upon a “bumper containing cellular material, e.g. solid foam”. While subclass 19/22 should probably be reviewed in

its entirety, the other two subclasses (19/03 and 19/18) will not be as relevant to the inventive concept. Instead those two subclasses are searched using the Boolean operator “AND” along with the term “foam” to garner a higher precision search [Example: (B60R19/03 OR B60R19/18) AND “foam”]. This allows the classification scheme to weed out the soccer shin guards and helmets, which may use the same terms as this bumper invention.

**How Is Citation Analysis Used** When a patent of relevance is located, it does not stand on an island by itself. Like the vast majority of advances in science and development, a patent is a baby step forward. By reviewing the art cited within the prosecution history of the patent (back citing), one can see the baby steps that preceded a particular improvement. Likewise, by reviewing all patents prosecuted after the patent of relevance (forward citing), one can see the baby steps that proceeded from a particular improvement. Performing this operation in a sideways manner (a forward cite followed by a back cite, or a backward cited followed by a forward cite), one can locate parallel art to the patent of relevance.

**How Are Iterations Used** An initial keyword-based search should be performed to learn about proper classification areas. Classification areas must be reviewed to learn new terms within the art. Forward and backward citation must be performed to learn both new terms within the art and new classification areas. Broad classifications combined with keywords must be performed to locate art that may not have been properly placed in a subclass. Further keyword searching should be performed to locate art outside the proper classification areas entirely.

It is important to note that all algorithms must be tried. Only after doing so will the most efficient algorithm be identified (similar to the identification of mathematical benchmarks by Wolpert and Macready [36]). At that point, additional resources may be assigned to the more efficient algorithms. Additionally, algorithms outside the basics identified above may be pursued depending upon the technology and nature of the information to be retrieved. These could encompass inventor searches, assignee searches (owner of the patent) and geographic searches (for example, looking for pachinko machines should probably focus on Japan).

With these basic algorithms for identifying relevant documents summarized, it must be noted with large and bold letters that image retrieval is the key. While one may start with general algorithms and carefully make algorithms more efficient through iterations, one can only identify the information of relevance through the use of rapid image retrieval.

A simplified example of the search process that would take a professional patent searcher 6 to 8 hours (this is using an engine with no image retrieval delays—for example, a Patent Office in-house system) is shown in Table 1.4.

## 1.8 Conclusion

Patent search, analysis and monitoring are business-critical, yet very time-consuming tasks that are performed primarily by manual means. A proper search methodology

**Table 1.4** An example of the search process for device/engineering drawings

Step	Action	References manually reviewed
1	Simple keyword search limited to titles or abstracts	300
2	Class/Subclasses combined with focused keywords	700
3	Forward and Backward citations searches	500
4	Medium complexity keyword search	1000
5	Class/Subclasses combined with looser keywords	300
6	Class/Subclasses in their entirety	500
7	Highly complex keyword search	500
8	Additional Forward and Backward citation searches	500
9	Class/Subclass ranges combined with focused keywords	500
10	Final Forward and Backward citation searches	200
	Total =	5000

will include usage of the major search mechanisms outlined above, will be well planned in advance, will exhaustively leverage the information collections appropriate for the search, and will use a constantly iterative approach. In this chapter, we have attempted to describe the practical experiences in and requirements for effective searching, analysis, monitoring, and overall management of patent information, from the perspective of patent information professionals. While databases and tools have long been supporting this process, advanced technologies are emerging to address age-old issues such as database quality as well as tackle new challenges [38]. These new challenges include:

- traditionally neglected issue of multilingualism and increasing volume of patent applications;
- wider variety of users from different backgrounds with various interests ranging from scientific and legal to business;
- expansion of patent information use to explore new technical and business opportunities in addition to the traditional IP protection approaches.

We hope that this chapter has contributed toward understanding of the current searching practices, systems and tools that would help in the further development of emerging retrieval technologies to assist the user in patent search, analysis and information management processes.

While new technology tools may greatly advance the patent search, analysis & monitoring processes, it is important to be reminded that tools simply assist and cannot replace the human mind. A good patent searcher is knowledgeable not only about the intricacies of different types of patent searching, but also the changing requirements of international patent laws, technical innovations and current developments in the different types of information sources available. The central role of the patent searcher continues to be essential in balancing the search requirements for recall and precision and for insightful analysis of the results.

The chapters following will investigate many of the topics addressed in this introduction in addition to many more to provide a comprehensive cross section of the many challenges patent information retrieval faces today.

## References

1. United States Patent & Trademark Office (1977) Eighth technology assessment and forecast report. Section II: 37. <http://www.ntis.gov>. Accessed 10 July 2010. NTIS Order Number PB 276375
2. Yang Y et al (2010) Enhancing patent landscape analysis with visualization output. *World Pat Inf* 32(3):203–220
3. Yang Y et al (2008) Text mining and visualization tools—Impressions of emerging capabilities. *World Pat Inf* 30:280–293
4. van Staveren M (2009) Prior art searching on the Internet: Further insights. *World Pat Inf* 31:54–56
5. Hantos S (2010) Helping others acquire, license, or invest in patents with confidence—A guide for patent searchers to patent due diligence. *World Pat Inf* 32(3):188–197
6. Simmons E (2001) Patents. In: Armstrong CJ, Large JA (eds) *Manual of online search strategies*, 3rd edn. Gower, Aldershot
7. Adams S (2006) Information sources in patents. In: McIlwaine IC et al (eds) *Guides to information sources*. De Gruyter SAUR, Munich
8. Simmons E (2006) Patents (literature), 5th edn. Kirk-Othmer encyclopedia of chemical technology, vol 18, pp 197–276
9. Alberts D (2008) The ever-changing role of information professionals in pharmaceutical R&D. *World Pat Inf* 30:233–237
10. Adams S (2000) Using the international patent classification in an online environment. *World Pat Inf* 22(4):291–300
11. Adams S (2001) Comparing the IPC and the US classification systems for the patent searcher. *World Pat Inf* 23(1):15–23
12. World Intellectual Property Office. <http://www.wipo.int/classifications/ipc/en>. Accessed 10 July 2010
13. Patent US Trademark Office. <http://www.uspto.gov/web/patents/classification/selectnumwithtitle.htm>. Accessed 10 July 2010
14. European Patent Office. [http://v3.espacenet.com/eclsrch?&locale=en\\_ep&classification=ecla](http://v3.espacenet.com/eclsrch?&locale=en_ep&classification=ecla). Accessed 10 July 2010
15. Schellner I (2002) Japanese file index classification and F-terms. *World Pat Inf* 23:1977–2001
16. Adams S (2010) The text, the full text and nothing but the text: Part 1—Standards for creating textual information in patent documents and general search implications. *World Pat Inf* 32:22–29
17. Adams S (2010) The text, the full text and nothing but the text: Part 2—The main specification, searching challenges and survey of availability. *World Pat Inf* 32:120–128
18. Hunt D, Nguyen L, Rodgers M (2007) *Patent searching. Tools & techniques*. Wiley, Hoboken, pp. 72–74
19. Chemical Abstracts Service (CAS) Registry. <http://www.cas.org/expertise/cascontent/registry/index.html>. Accessed 10 July 2010
20. Derwent World Patent Index. [http://thomsonreuters.com/products\\_services/legal/legal\\_products/intellectual\\_property/DWPI?parentKey=442831](http://thomsonreuters.com/products_services/legal/legal_products/intellectual_property/DWPI?parentKey=442831). Accessed 10 July 2010
21. IFI Claims Patent Services. [http://www.ificlaims.com/searchaids\\_chemterms.html](http://www.ificlaims.com/searchaids_chemterms.html). Accessed 10 July 2010
22. Austin R (2001) The complete Markush structure search: Mission impossible? PIUG Northeast Workshop. [http://www.stn-international.com/uploads/tx\\_ptgsarelatedfiles/piug1.pdf](http://www.stn-international.com/uploads/tx_ptgsarelatedfiles/piug1.pdf). Accessed 10 July 2010

23. Simmons E (1991) The grammar of Markush structure searching: Vocabulary vs syntax. *J Chem Inf Comput Sci* 31:45–53
24. STN Structure Searching Cluster Databases. <http://www.cas.org/support/stngen/clusters/structure.html>. Accessed 10 July 2010
25. Questel Merged Markush Service. [http://www.questel.com/Prodsandservices/mms\\_chemistry.htm](http://www.questel.com/Prodsandservices/mms_chemistry.htm). Accessed 10 July 2010
26. Derwent World Patent Index, op.cit
27. IFI Comprehensive Database. <http://stneasy.cas.org/dbss/help.IFICDB.html>. Accessed 10 July 2010
28. National Center for Biotechnology Information BLAST Glossary: <http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/glossary2.html>. Accessed 20 July 2010
29. Falciola L (2009) Searching biotechnology information: A case study. *World Pat Inf* 31:36–47
30. Andree PJ (2008) A comparative study of patent sequence databases. *World Pat Inf* 30:300–308
31. Sheiness D (1996) Patenting gene sequences. *J. Pat. Trademark Off. Soc.* 78:121–137
32. Brown J (2010) STN international presentations on databases and products. [http://www.stn-international.com/fileadmin/be\\_user/STN/pdf/presentations/res\\_IPsearching\\_0806.pdf](http://www.stn-international.com/fileadmin/be_user/STN/pdf/presentations/res_IPsearching_0806.pdf). Accessed 17 July 2010
33. Yoo H et al (2005) Intellectual property management of biosequence information from a patent searching perspective. *World Pat Inf* 27:203–211
34. Yoo H op.cit
35. GenomeQuest Search Strategies (2009) In: GenomeQuest user manual, 5.2 edn
36. Wolpert DH, Macready WG (1995) No free lunch theorems for search. Technical Report SFI-TR-95-02-010. Santa Fe Institute
37. DeMarco D, Davis A (2010) Mechanical patent searching: A moving target. In: PIUG 2010 annual conference. <http://demarcoip.com/downloads/2010-DeMarco-PIUG.pdf>. Accessed 20 June 2010
38. Bonino D, Ciaramella A, Corno F (2010) Review of the state-of-the-art in patent information and forthcoming evolutions in intelligent patent informatics. *World Pat Inf* 32:30–38



# Chapter 2

## An Introduction to Contemporary Search Technology

Veronika Stefanov and John I. Tait

**Abstract** This chapter is the counterpart of the preceding chapter. It gives an overview of some of the most important terms and concepts used in Search Technology and Information Retrieval today. We hope it can be useful to readers who are not researchers in these areas. After a short dip into the history of the field, we start with a high level overview of the different types of search, and the gap between user requirements and how search systems can be evaluated, finally narrowing it down to the main evaluation methodology used today. This is followed by a step by step guide to the architectural components of a generic full text document search system and its design implications. We then describe how the underlying models define to a large extent what the system can and cannot do. This chapter concludes with a short introduction to semantic search and an outlook to the challenges in patent IR, the main subject of this book.

### 2.1 Search Technologies and Information Retrieval

We have called this chapter “Introduction to Contemporary Search Technology” rather than, for example “Introduction to Information Retrieval” because the subject of Information Retrieval as a whole is very broad and much of it is of little or no interest to those involved with practical patent search.

Information Retrieval might be defined as the science and technology of searching for and accessing information in documents, or in parts of documents. This definition by Manning et al. [21] can be useful to see the core, as well as the whole area:

Information Retrieval is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).

---

V. Stefanov (✉) · J.I. Tait  
Information Retrieval Facility, Vienna, Austria  
e-mail: [v.stefanov@ir-facility.org](mailto:v.stefanov@ir-facility.org)

J.I. Tait  
e-mail: [john.tait@ir-facility.org](mailto:john.tait@ir-facility.org)

It is a classic scientific endeavour, with both theoretical and experimental (or perhaps better: empirical) branches, and a well established scientific paradigm subscribed to by many in the field, whether in academia or the industry. The scientific aspect has always (since the 1940s or 1950s) been closely related to a practical engineering aspect, which seeks to deliver operational systems. This has very wide use and implications through, for example, well-known internet search engines like Google and Microsoft's Bing.

In other areas, there is a well recognised terminological distinction between the science and technology aspects: for example between physics and mechanical or electrical engineering. As is all too often the case in computing, unfortunately Information Retrieval is a portmanteau term for both the scientific and engineering aspects. Since the practical patent searcher mainly needs an understanding of the technological or engineering aspects, we have chosen the narrower term "search technology" for this introductory chapter.

This is not say the practical patent searcher might not find the science of Information Retrieval useful: in particular over the years many empirical studies in IR have presented surprising results about which technologies are the most effective for search. These results help practitioners reflect on, and improve their practice. Information Retrieval owes its origins to 19th Century Library Science (see e.g. Schrettinger [31]), but was inspired and transformed by the development of computerised "mechanised" information systems after World War II, and perhaps especially by Vanevar Bush's prescient Memex article [8]. The first use we can find of the name Information Retrieval is by Mooers in 1950 [22] who was, incidentally an early and vocal critic of the use of Boolean Logic (as opposed to ranking) in search technology systems [23]. Boolean retrieval, despite this, remains the mainstay of practical patent search.

It is worth noting that the early systems would invariably be described today as searching metadata: typically, the data they searched were author, title, some index terms or keywords, and perhaps an abstract: it was only in the 1960s or even 1970s it became common place to analyse and index (and therefore make searchable) the full text of the document. As noted above, Information Retrieval has a strong theoretical tradition. But in practice the field is principally driven by experimental work. Prime amongst this was the work by Cleverdon and others at the Cranfield Institute of Technology in England in the 1960s [10, 11], which continues to be influential (see also Chap. 3 in this volume).

It is worth noting that patent search has been an application of interest from the earliest days of information retrieval, although in the early literature it is sometimes difficult to distinguish between computerised Information Retrieval and the use of older, mechanical sorting and selection devices, like card sorters.

A complete survey of search technology, let alone Information Retrieval as a whole, goes beyond the scope of this chapter (or the whole volume). There are a wide range of text book introductions to the subject ([9, 21], etc.), although generally they nowadays focus on web search.

By way of introduction to the rest of the chapter, a couple of points are worth making, which might surprise patent searchers.

First it is the accepted wisdom of the Information Retrieval research community, based on a significant body of experimental evidence accumulated over many years, that ranked retrieval systems are more effective than Boolean or other structured query systems. Systems experiments almost always show the familiar set-based Boolean retrieval systems are less effective than systems which present the searcher with lists of potentially relevant documents ordered with the most likely to be relevant first, then the next most to be relevant, on so on towards progressively less relevant documents. We will return to this later in the chapter.

Second, a recent insight has been that retrieval from very large (web scale, petabyte scale) collections of documents may be different in kind from retrieval from smaller scale collections. The reasons for this remain unknown at the present time, but may be the result of the pervasive nature of phenomena matching Zipf's Law (see [3] and Chap. 12 of this volume for more).

As noted above, experimental work is the hallmark of IR research. Therefore, rather than diving in to the technology, in the next section of this chapter we give a brief introduction to IR evaluation.

## 2.2 Finding a Search Technology that Works for You

There are many ways of looking at the variety of IR systems: from an information theory perspective, a historical perspective, a systems engineering perspective, an IR researcher's or a librarian's perspective. We begin with the final purpose for which such systems are designed: searching and finding what you are looking for.

Which techniques and tools are useful greatly varies with the type of search task. Searching can take many different forms. One way of structuring them could be the following types [24, 28]:

**known-item search** The user is searching for an information object which is already known to them; also known as direct search.

**exploratory search** The user is seeking to learn about a topic but does not know in advance what may be important.

**browsing** The goal is unclear, the user is not sure whether or how the requirements can be met.

**exhaustive search** The user is trying to learn everything about a particular topic.

Only the first type, known-item search, is more or less supported by classic search engines and classic search engine evaluation. The information need is well defined and can be expressed by the user (subject terms are known, maybe also the author, document type, creation date, etc.), and the correct answer may be found with no or very few iterations.

In an exploratory search, the user cannot provide an exact query at the beginning, and so will be confronted with a large amount of potentially interesting results. The approach will be very iterative, with the user needing to review the results of every step to refine the query, gradually learning about the topic. Additional functionality

such as aggregation and visualisation of search results, as well as automated query rewriting, can support the user.

Browsing can take advantage of links inherent to the document collection, such as weblinks or citations. If users have a fairly good idea of what they are looking for, following link pathways allows them to refine their perception of their information need.

Exhaustive search enjoys only very limited support from any existing IR system. The requirements on both the users (in terms of background knowledge) and the system performance are high. Nevertheless, exhaustive searching is an everyday requirement in many domains (law, patents, medicine, intelligence).

### ***2.2.1 Can You Choose the Best IR System?***

How can you choose an IR system for your tasks? How do you test and compare? The first thing most people probably do is to give a system some test queries and look at the results. But how can you judge whether these documents are the best matches in the whole collection? The unfortunate answer is that the only way to really know, would be to look at all documents in the whole collection and check. For any meaningful IR situation this is not feasible (if the collection is small and you know all the documents in it, you do not need a retrieval system. . .).

One way of looking at it could be to assume that if many people use many systems over a long time, certain trends might become apparent. They might stop using less useful systems and switch to the systems that save them time and effort, meaning that users actually vote by their usage.<sup>1</sup> But is the most widely used system also the best? How do users choose?

Professional users have to choose from tools they might buy, and this can be done with trial licenses or calls for bids to system vendors. They also often do not have a real choice (lack of information, prohibitive switching costs from vendor lock-in, licensing, knowledge/training investment, etc.). As criteria for selecting patent search tools for example, the data coverage, document delivery, import and export functions as well as the company behind the tool are equally important, if not more than the pure retrieval effectiveness [15].

For some systems, it is possible to infer user happiness/searcher trust/usefulness metrics (see also Chap. 20) from secondary values. Ad revenues, e-commerce deals or measures of returning users can be meaningful for web-based applications, whereas enterprise search solutions try to measure productivity gains.

Compared to researchers, professional users know their use cases intimately. They can focus on just their own needs and ignore all other issues, which in turn allows them to select tools for their work.

---

<sup>1</sup>This seems to have happened in the late 1990s between Web search engines. Those that viewed web sites as plain text documents were replaced by search engines that used the links between sites to choose those that were most likely more useful to more people. The quality of the search results using link analysis was so much better that people switched.

### ***2.2.2 User Knows Best: User-Centred Evaluation***

The overall test of a system is the usefulness to its users. User-centred evaluations can and are being done, but they are expensive and difficult to do correctly for a number of reasons [17]:

1. a large, representative sample of actual users is needed
2. each system must be equally well developed and must have a user interface
3. each participant must be equally well trained on each system
4. the learning effect must be controlled for

Because of these issues, real user-centred evaluation is rare, which has led to a certain unfortunate lack of communication and feedback between IR researchers and those who might potentially use their search systems. Some specific ways forward for patent search are explored in Trippe and Ruthven's chapter in this volume (Chap. 6).

### ***2.2.3 Laboratory Tests: The Cranfield Model***

Already in the early days of computer-based IR, researchers devised testing methods that can be likened to laboratory tests. They ignore a large amount of the "noise" and ambiguities of real use cases, and allow for empirical, reproducible tests that yield quantitative results on large amounts of data. The so-called Cranfield tests lead to the main evaluation methodology still used today. The second part of this book discusses evaluation methodologies within the Cranfield paradigm in more detail.

For such an evaluation, the following items are needed:

- a suitable collection of documents
- some (representative) queries on this collection
- and for every query, a list of documents (ideally a complete list) that are relevant (the relevance judgements)

Given this information, automated tests can be run that compare the actual results to the target results and quantify the differences.

The first requirement, a suitable collection of documents, which has to be large enough to make the statistical results meaningful, can already be difficult to achieve. But the second and especially the third requirement are the real hurdles to large scale testing.

How are the relevance judgements produced? The gold standard is manually judged results, for which a large number of highly skilled and motivated judges is needed. Ideally, all relevant documents are collected beforehand for all test queries. Apart from the fact that for many queries there simply is no "right" answer, research has also shown that human judges tend to disagree in what they find relevant [30].

Additionally, new test queries have to be found for every competition to ensure fair conditions as well as to avoid over-optimisation of the systems to the training

set. It is not surprising that many approaches have been developed over that last decades that are able to create relevance judgements automatically from the collection and the queries. These values are tainted with uncertainty, but are still useful for many types of evaluation, which otherwise could not be performed at all. Chapters 4 and 5 describe how in the area of patent information retrieval evaluation, the citations contained in the documents can be used to obtain usable relevance judgements.

### 2.2.3.1 Evaluation Conferences

The Cranfield paradigm forms the basis for a number of longstanding evaluation conferences, where the organisers provide the data, queries and relevance judgements. The efforts of TREC,<sup>2</sup> CLEF,<sup>3</sup> NTCIR,<sup>4</sup> and FIRE,<sup>5</sup> have improved the situation of IR evaluation greatly by providing researchers not only with urgently needed data and frameworks, but also with a community and comparable research [25]. Within this volume one can find more about activities focused on patent search in Chaps. 3, 4, and 5. It remains to be seen how recent complementary efforts such as PatOlympics<sup>6</sup> can foster the interaction between creators of retrieval systems and information professionals.

## 2.2.4 Quantifying the Difference: IR Measures

Assuming that the three evaluation requirements above have been taken care of, and that the experiments have been conducted, how should the difference between the actual results and the target results be analysed? It helps to know the goals of the system to be able to select substantive values. The Cranfield tests established desirable characteristics of an IR system—precision and recall—which are at the heart of every IR evaluation.

### 2.2.4.1 Precision and Recall

Precision looks at how many “wrong” documents were caught together with the right ones, while recall looks at how many “right” documents were missed. Both

---

<sup>2</sup>Text REtrieval Conference (TREC), <http://trec.nist.gov/>.

<sup>3</sup>Cross-Language Evaluation Forum (CLEF), <http://www.clef-campaign.org/>.

<sup>4</sup>NII Test Collection for IR Systems (NTCIR) Project, <http://research.nii.ac.jp/ntcir/index-en.html>.

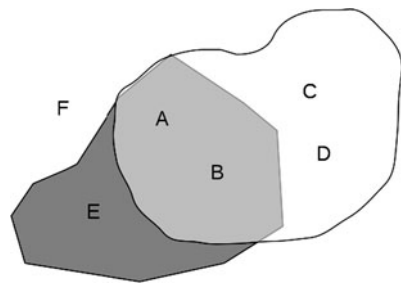
<sup>5</sup>Forum for Information Retrieval Evaluation (FIRE), <http://www.isical.ac.in/~clia/>.

<sup>6</sup>PatOlympics Interactive Patent Retrieval Competition, <http://www.ir-facility.org/events/irf-symposium/irf-symposium-2011/patolympics>.

**Table 2.1** Precision is  $0.\overline{6}$ , Recall is 0.5

Relevant documents	Retrieved documents
A	A
B	B
C	E
D	

**Fig. 2.1** Illustrating Table 2.1: Precision compares the overlap to the whole *dark area*, recall compares the overlap to the whole *white area*



are numbers between 0 and 1 (often expressed as percentages), where 1 is best.

$$Precision = \frac{\text{number of relevant items retrieved}}{\text{number of items retrieved}}, \tag{2.1}$$

$$Recall = \frac{\text{number of relevant items retrieved}}{\text{number of relevant items}}. \tag{2.2}$$

A precision of 0.8 means that for every four correct documents in the result list there is one mistaken one that is not relevant to the query. A recall of 0.8 on the other hand tells you that the result list contains only 80% of all the documents that should have been retrieved.

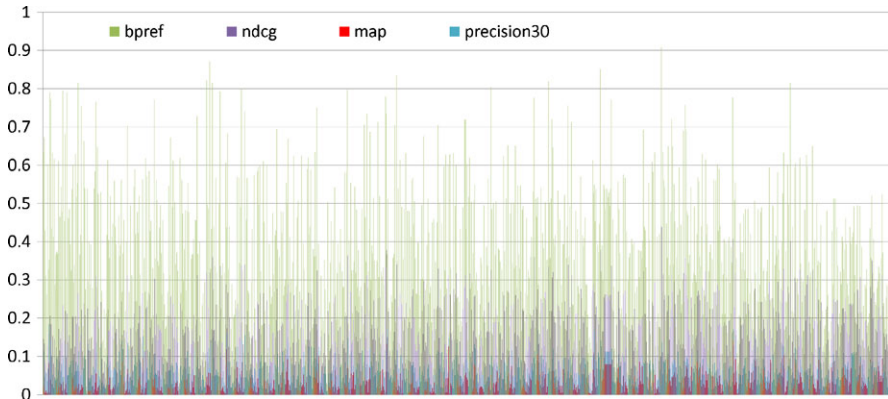
You can view them as measures of false positives<sup>7</sup> and false negatives.<sup>8</sup> They only make sense together, as it is trivial to increase just one of them,<sup>9</sup> and as illustrated in Fig. 2.1 they are usually contradictory.

For most systems, it is generally unknown which levels of recall and precision they can achieve. For commercial search tools, no published evaluations exist. And for the academic systems that are submitted to evaluation conferences, the results must be taken with caution. It lies in the nature of the Cranfield paradigm that the absolute values of the evaluation measures are not meaningful by themselves. They can only be used to compare different runs on the same test setup. Unfortunately this also means that the results obtained at an evaluation conference in one year cannot

<sup>7</sup>Also known as type I error or  $\alpha$  error.

<sup>8</sup>Also known as type II error or  $\beta$  error.

<sup>9</sup>How do you reach a “perfect” recall of 1.0? Put the whole collection in the result set. How do you achieve a very high precision? Limit the result set to just a few documents.



**Fig. 2.2** Results by query (raw data from the TREC-CHEM 2009 prior art task)

be directly compared to the previous year, although all the major evaluation campaigns have maximising reproducibility as a goal [5, 6]. See Chap. 6 for a critique of the recall/precision model.

#### 2.2.4.2 Beyond Precision and Recall

Recall and precision work on sets and have no notion of ranking [32]. Since ranked result lists are a common feature of search systems and the quality of the ranking greatly influences the quality of the result for the users, derived measures had to be found. A commonly used measure is the mean average precision (MAP). The precision value is different, depending on how far down the result list you look. Average precision is the average of all precision values at the point where each relevant document is placed in the ranked list. MAP is the mean of the average precisions for a group of queries, since the values can depend heavily on the queries. In fact, as can be seen in Fig. 2.2,<sup>10</sup> the raw data obtained in large scale experiments are, in most cases, difficult to interpret.

Additionally, since many use-cases favour recall over precision or vice versa, metrics matching these requirements can be used. For a stronger focus on precision, metrics that only look at a smaller amount of documents at the top of the list are useful, whereas for recall-oriented cases, it can help to measure the precision at a given level of recall, which would indicate how many wrong documents the user will encounter before the desired recall is reached. Chapter 3 of this book contains examples of more advanced measures.

Interested readers taking a look at the proceedings of IR conferences and evaluation workshops will find advanced charts and tables comparing these measures [20, 27]. It is usually not intuitively understandable what the results “mean” for every day search tasks. Compared to the types of tasks outlined above, the Cranfield

<sup>10</sup><http://www.ir-facility.org/research/evaluation/trec-chem-10>.



type evaluations do not represent iterative or complex searches. It is possible to evaluate individual supporting methodologies, such as for example query rewriting methods, by comparing the results of the original query to the modified one, but anything that resembles users extracting information from one result and applying it to the next query while using information from a third source cannot be represented in this model, although there have been recent attempts to overcome this problem [7].

### 2.2.5 System Characteristics

Apart from the result list, other characteristics of an IR system can be measured in a straightforward way [21], as for example:

- latency of showing results (as function of index size) in seconds from submitting a query
- collection size and how is it distributed over topics in megabytes or documents
- timeliness of the data in the collection): what is the latest date at which the index is guaranteed to match the current version of a document

For users, the query interface and query languages are very important, but their features cannot be captured so easily:

- expressiveness of the query language (languages can only be “measured” in term of feature checklists)
- performance (speed) of complex queries (as opposed to retrieval latency due to index size)

## 2.3 System Components and Architecture

If you wanted to build your own IR system, how would you do it? As different as they might appear on the outside, most systems follow a similar overall architecture.

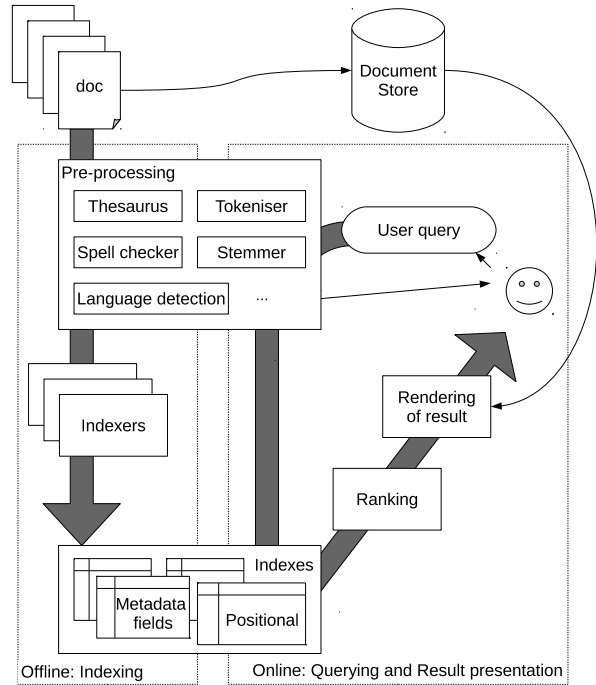
Contrary to how it is displayed in movies and on TV, or how simple desktop file search tools work even now, IR systems do not start to scan all documents when you submit a query. Instead, most of the work is performed before, at index time (“offline”), and only some tasks are performed live at query time (“online”). Depending on the use case, it makes sense to do more or less offline. Such design decisions make or break a successful IR system.

In general, a system will have the following components:

1. Indexing
2. Querying
3. Result presentation

The following gives an overview of the main steps, the purpose and the challenges of each part, as summarised in Fig. 2.3.

**Fig. 2.3** Overview of the components of a search system, cf. [2, 21]



### 2.3.1 Indexing

Indexing means preparing a second, separate representation of the documents, optimised for retrieval.<sup>11</sup> If we assume that the user will be searching with terms as queries, and will want to get all the documents that contain these terms, it would be useful to have a list of the terms and for each term a list of the documents where it occurs. This simple list is the so-called *inverted index*, also known as postings list. It is the most basic index. The bi-directional mapping between terms and documents is known as *term-document matrix*. Such a representation loses the order of the terms in the document, so if the users are to be able to search for multi-words or phrases, or use positions of the terms with wildcards, the index will have to store this information as well [2].

The index can only give the information that is stored in it, so if the result presentation should contain a snippet of the document with a highlighted search term, this has to be taken into account and the necessary data stored in the index. For ranked result lists, the information needed for weighting of terms and documents also has to be included in the index somehow. See Sect. 2.4.2.3 of this chapter for examples.

<sup>11</sup>Before indexing, it is necessary to get the documents (with web crawlers, fetchers, etc.), which can be a challenge in itself.

### 2.3.1.1 What Is a Term?

Indexing based on the terms in a document collection requires a working definition of “term”. Tokenisation is the process of splitting text into individual words or terms. The straightforward approach of splitting at spaces and punctuation marks can lead to problems with numbers, URLs or acronyms, so more advanced rules with exceptions, word lists, and thesauri can greatly improve the performance of the final system. Tokenisation is also language specific and benefits from linguistic knowledge: in many Asian languages no spaces are used to separate words, which requires advanced methods of word segmentation, but also German or Dutch compound nouns require a compound splitter.

For languages with inflection, stemming is often the next step. Stemming removes suffixes from the terms, reducing them to their core or stem. This process is also language specific, and while it loses a significant amount of information contained in the endings (distinctions between plural and singular, verb and noun, past and present), it makes querying easier, as the user does not have to enumerate all possibly matching variations in the query string.

Terms that occur in practically all documents and many sentences, such as “of”, “the” or “and”, called stop words, can be removed from the index, as they do not add any discriminative information that could improve the results, while making up a large portion of the size of the index. When it comes to phrase search, the missing stop words have to be taken into account, either by removing them from the phrase to be searched, or by making sure that the index used for the search still has them.

Further processing of the word list might include checking and treating spelling, OCR or transcription errors [18].

### 2.3.1.2 Fulltext, Metadata and Other Information

A lot of content is actually of a semi-structured nature. It contains unstructured parts such as text or images, which have to be prepared to become searchable, as well as structured content such as dates and other document metadata, which lend themselves much more easily to searching (e.g. “all PDF documents created between February 7th and 10th”). If these values should be searchable together, this integration must also be prepared at the indexing step. The same is true for any additional enhanced search methods, such as semantic information extraction of events and relationships. In fact, it is not uncommon to create separate indices for these different types of data and query them all in parallel at query time.

### 2.3.1.3 System Characteristics and Engineering Decisions

Other important distinguishing features of indexers are the indexing speed (in documents or Kilobytes per second), the resulting index size (compared to the original documents), and whether the index can be easily updated when there are changes in

the collection, or whether it has to be completely recreated. Querying speed depends on how easily the index can be accessed, and that depends on the physical instantiation of the index: it can be a single large file, a collection of files, a database, and stored on one machine or distributed.

### 2.3.2 Querying

The querying component consists of a query parser and whatever tools are necessary to match the user's query to what is contained in the index. A free text query is treated similarly to the documents in the indexing step: it gets tokenised and stemmed. "Did you mean ..." suggestions can be created by performing a spellcheck on the query or by comparing it to a list of frequent queries.

Users are generally not able to construct perfect queries. They might get close for known-item searches, but for all other types of search, they simply cannot know beforehand. The search system can support them with automated query rewriting. Users often come across concepts that can be expressed in many different ways, where they cannot know which one will lead them to the desired results. On the other hand, many words have more than one meaning, which is clarified by the context of a sentence or paragraph, but remains ambiguous when used in a query.

A thesaurus can be used to improve the situation by automatically adding synonyms to the query. But since the terms in the original query lack context, this will typically lead to much less precise queries, as terms from unrelated domains are added to the query.

Since not even the most advanced algorithms will know more about the domain context of the query than the user, another method is to perform an initial search and then ask the user directly for feedback to the retrieved documents. The user marks a few of the top retrieved documents as (non-)relevant, which makes it possible to automatically modify the query in a way that finds more relevant documents, much the same way in which users would modify their queries to include and exclude items after seeing the first results. This type of relevance feedback has been used since the 1960s and is known to be effective [26, 32].

Sometimes user feedback is not available, so in the 1990s, pseudo-feedback was invented. It assumes that the top documents retrieved initially are close enough to the intended result, so that related terms can be taken from these documents to create a second, improved query. The modified query contains related terms and synonyms to the original query terms. The result of the second query is presented to the user. This approach is also known to be effective, in particular for short queries [32].

After the documents have been retrieved, if the result list is to be ranked, the retrieved documents have to be scored by whichever model the IR system uses (see Sect. 2.4.2 of this chapter for ranking models). If the result list is constructed from different sources, they have to be merged into one uniform result with one overall ranking before being presented to the user.

Machine learning has been successfully applied to ranking: The systems learn a ranking function based on a ranked training data set [19]. In the early 2000s,

such techniques became dominant at commercial search engine providers, whose expertise is visible in the results of challenges such as the recent “Learning to rank challenge”.<sup>12</sup>

### 2.3.3 *Result Presentation*

The linear result list is a very common and simple presentation mode. It can be sorted and filtered by the available metadata of the documents (e.g. date, size, file type). If the underlying model does not support ranking, sorting the documents chronologically, for example, can be very useful.

The requirements for the presentation of results depend heavily on the domain. Web search engines have evolved to provide snippets from the pages as well as summaries, direct links to parts of the pages found, maps, or images, as this saves searcher’s time. Patent or legal searches value depth of knowledge more than time and will not be satisfied with snippets or summaries alone but will want easy access to the full document, with highlighting of query terms or other indications of why this document is in the result list.

Result snippets and summaries can be static (independent of the query) or dynamic. The static ones can be created and stored at indexing time, whereas creating dynamic summaries at query time may require access to the full document or elaborate calculations and can be a costly operation. They can help to explain why the document was retrieved for the particular query.

There are many quite sophisticated summarisation approaches in the area of Natural Language Processing. A simple method is to show the search term surrounded by the words that precede and follow it in the text, which is called keyword-in-context, or KWIC. The context can be a fixed window or adjusted to sentence boundaries with linguistic methods [21, Sect. 8.7].

For document collections with metadata that can be seen as a network, such as academic publications, a visualisation of the network graph can be useful (for example, MS academic search uses people who have published together), whereas geographic metadata can be visualised on maps<sup>13</sup> (see Chap. 11).

Other options include word clouds that show words occurring together (in a document, in a group of documents, in the search results), and words that occur more often larger<sup>14</sup> than less frequent ones.

For browsing or explorative searches, faceted search can be very useful to get an overview over a large result set. Facets are attributes of the documents, either given as metadata or computed on the fly with clustering or classification algorithms. They are well known from the user interfaces of online stores (where you can filter the

---

<sup>12</sup>Yahoo! Labs Learning to Rank Challenge <http://learningtorankchallenge.yahoo.com/>.

<sup>13</sup>For example, Freepatentsonline shows inventors’ addresses on a map at <http://www.freepatentsonline.com/maps>.

<sup>14</sup><http://www.wordle.net>, <http://deeperweb.com>.

thousands of available shoes by colour, size, manufacturer, material, etc.) and act as a kind of drill-down into “regions” of the result set. They are a convenient alternative to complex search forms with multiple fields because they can be used on demand after the query returns, and only until the result set is small enough for browsing.

## **2.4 IR Models**

The ways users can express their information needs as queries and how the queries are used to find the desired documents depends on the model built into the system. Indexes, query parsers and ranking components work based on assumptions of what documents and queries are. In the past, larger increases in the performance of IR systems of the past have been related to paradigm changes in IR models used.

### ***2.4.1 Boolean IR***

The boolean model is the simplest IR model. It is clear and precise: A document either matches or it does not match the query. The user is in control and has transparency over what is retrieved. The search terms are linked with boolean operators: AND, OR and NOT. Using AND greatly increases precision and lowers recall, whereas OR quickly lowers precision and increases recall.

Boolean systems are generally good for expert users with clear understanding of their needs and the collection, as it requires a lot of skill to come up with a manageable number of hits. In the basic version of the boolean model, all terms are weighted equally, so it can be quite challenging to find the sweet spot between a huge result set with too many documents and an (almost) empty result set.

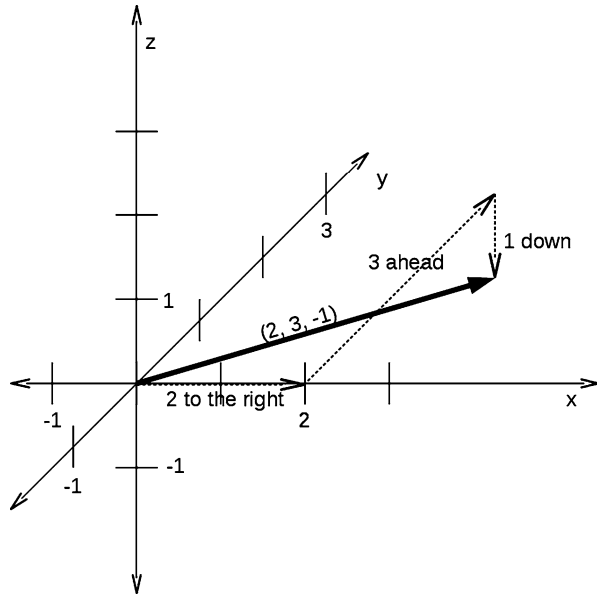
In extended versions, term proximity operators and wildcards can be used. Boolean operations are set operations on a set of documents, which implies that the results cannot be ranked. In practise, using some of the document metadata to display the documents in an order, for example, chronologically, can work very well for many applications.

Boolean systems have dominated commercial tools for decades. In the 1990s, Turtle [33] first showed that free text queries performed better than expert boolean queries on a legal document collection.

### ***2.4.2 Ranked IR***

If you can find a way to display the best fitting documents at the top of the list, this solves the Boolean systems’ problem with the result set size. Almost all contemporary search technologies are based on ranked retrieval, and it is the accepted wisdom

**Fig. 2.4** A vector in 3-dimensional space:  $(2, 3, -1)$



amongst the IR community that ranked retrieval is almost always more effective than Boolean Retrieval.

Ranked retrieval needs a scoring formula that can provide a numeric value of how likely a document is useful to the searcher, of how well it matches the query. This property makes it possible to “narrow” or “broaden” a search.

### 2.4.2.1 Vector Space Model

A vector is a geometric object representing a direction.<sup>15</sup> It resembles a list of numeric values, one per dimension. The three-dimensional vector  $(2, 3, -1)$  represents the direction “two steps to the right along the  $x$ -axis, three steps ahead along the  $y$ -axis, and one step down along the  $z$ -axis” (Fig. 2.4). Any list of numbers, no matter how long, can be viewed as a vector. The corresponding space has as many dimensions as there are values in the vector, which might be unimaginable to humans, but mathematically it works just the same.

The vector space model uses vectors to represent documents and queries. The dimensions of the vectors should correspond to the distinguishing features of the documents, so if terms are what will be used for querying, then the vectors will have as many dimensions as there are unique terms in the collection. A document’s vector will contain a non-zero value at the slot of a term if the term occurs in the document, and zero if not [29]. These very large vectors are also very sparse, meaning that most

<sup>15</sup>Actually, a direction and a length, but the length is irrelevant for our purposes.

of their values are zero, a property which can be exploited in the implementation to improve the data model.

Models in which word order is fundamentally disregarded are known as “bag of words” models, and so the vector space model is a “bag of words” model. Note that this does not preclude using word order in queries: it is more an issue of efficiency.

In order to use such a system to score a document’s relevance to a query, the query is treated as a small document, and a vector is created for it at query time. The similarity between the query and a document is then assumed to correspond to some property of the angle—typically the cosine—between their vectors. This approach is very useful for ranking the search results because it can represent a continuous degree of similarity. The cosine for example is 1 for equal documents and 0 for documents that have no terms in common.

This technique works for all kinds of data that can be represented as vectors (images, music, network graphs, molecule structures), and is useful also for classification and clustering. Classification is inherently similar to retrieval, as it can be seen as classifying documents into the two classes *relevant* and *not relevant*. But because there are many more irrelevant documents than relevant ones, the distribution is very skewed, and using unmodified classification techniques directly for retrieval might run into problems [21].

As mentioned in the section about querying above, using the terms occurring in the documents directly for searching results in noise and ambiguities caused by synonyms<sup>16</sup> and polysems.<sup>17</sup> Latent Semantic Indexing (LSI) [13] is a strategy that uses matrix computation methods to resolve some of the problems caused by synonyms. A (computationally expensive) multi-step process on the term-document matrix finds a much smaller approximation to the original matrix that replaces the terms with “concepts”, grouping terms with similar semantics [14]. The method was patented in 1988 (US Patent 4,839,853 [12]). Latent Semantic Indexing is a statistical approach to detecting semantic information in unstructured text. Section 2.5 below focusses more on explicit semantic methods.

#### 2.4.2.2 Probabilistic models

Probabilistic methods are based on the idea that it is possible to estimate the probability of a term appearing in a relevant document if you have some known relevant and non-relevant documents. Probabilistic IR is somewhat similar to the approach taken with the vector space model, in that they are generally based on the bag of words approach, but it is resting on the sound foundation of probability theory. Probabilistic methods have been investigated in IR since the 1970s and won new support with probabilistic methods in computational linguistics in the 1990s, but never achieved the performance expected or hoped from them [21]. In the result list,

---

<sup>16</sup>Several words for one meaning.

<sup>17</sup>One word with several meanings.



the documents can be ranked by their probability of being relevant to the query:

$$Probability(\text{document is relevant to the query}|\text{document, query}). \quad (2.3)$$

### 2.4.2.3 Term Weighting: **tf.idf**

For the above models to work, some numbers have to be inserted. Which values should be used for the non-zero values in the term vectors? Zeros and ones, or how often the term occurs in the document? Should they be normalised in any way?

Typically, **tf.idf** or one of many values derived from it are used. **Tf.idf**, the ratio of term frequency to document frequency, reflects the searcher trying to find terms that are rare overall (discriminative) but frequent in the requested document. It is the “magic number” of Information Retrieval.

$$\frac{\text{term frequency}}{\text{document frequency}} = \frac{tf}{df} = \text{tf.idf}. \quad (2.4)$$

The document frequency (**df**) of a term is the number of documents in the collection in which the term occurs. Each term has one **df** for the whole collection. The term frequency (**tf**) of a term is the number of times a term occurs in a document. A term therefore has one **tf** per document in the collection.

If a term is rare throughout the whole collection, its **df** and **tf** are small and the **tf.idf** for all documents is similar. If it is rare overall but frequent in a single document, its **df** is still small, but the **tf** for that document is large, making the **tf.idf** larger for that term in that particular document. Work has continued to provide improved formulae within this framework, most notable of which is **BM25F** [34].

All similarity depends on the keywords, so this approach is sensitive to vocabulary differences and the preprocessing of the documents (see Sects. 2.3.1 and 2.3.2 of this chapter). It assumes that the frequencies are independent, and disregards the order of the terms in the documents. It can be extended with phrase search, wildcards and (quasi-) boolean operators though.

Such search engines are independent of the type of data, as long as they can be accessed to be indexed (i.e., somehow turned into numbers). Many open-source search engines exist that are based on this approach; the most widely used is Apache Lucene.<sup>18</sup>

## 2.5 Semantic Search

Semantic technology is the subject of many hopes, as it may allow search systems to take (some of) the meaning of the words into account, as opposed to “just counting” them. If applicable to the domain and done successfully, it can be expected to

---

<sup>18</sup><http://lucene.apache.org>.

improve recall while keeping precision at least constant if not also increasing it [21]. The requirements consist of a suitable information representation and the ability to perform natural language processing.

In the patent search community “Semantic Search” is often taken to include Latent Semantic Indexing and related techniques. Generally within the IR community, and more so in the Semantic Web, Knowledge Management and Computational Linguistics community, it is generally considered a variant and extension of vector space models: hence its treatment in this chapter.

Knowledge bases (ontologies, thesauri, and taxonomies) represent concepts and relationships—usually within a subject area—that a community can agree on. They are used to classify, structure, define or represent, and have the additional value of aiding cross-language interoperability, and are often created for company-specific data. They can be used in semantic search for query expansion, searching by concepts instead of terms, as well as broadening or narrowing search.

**Controlled vocabulary/Glossary** A list of terms and definitions. Used to reduce the variability of terminology use.

**Taxonomy** A knowledge hierarchy where items are connected to each other by parent-child, part-of or instance-of relationships. Classification hierarchies like the International Patent Classification (IPC) are a kind of taxonomy.

**Thesaurus** A network of terms connected by hierarchical, equivalence or associative relationships. Synonym dictionaries used by patent searchers are a kind of thesaurus.

**Ontology** A taxonomy with multiple, precisely defined links between the items that represents knowledge as a set of concepts and their relationships. Different kinds of ontologies are suitable for different purposes (reasoning on the data, fuzzy search, etc.).

Information Extraction is the identification of facts from unstructured text, so that knowledge bases can be built with little or no human effort. It depends in part on Named Entity Recognition which uses lists of known multi-words (as found in dictionaries, thesauri, ontologies, taxonomies) to recognise entities such as places, organisations, persons and events in text documents [32]. Relation extraction finds the relationships between entities (e.g. *person [works at] organisation*). These feats can be accomplished with pattern-based, with statistical, or with hybrid methods. State of the Art systems have some ability to deal with previously unseen terms, and Named Entity Recognition has proved itself ready for deployment in industrial settings, like business intelligence. Given the prevalence in patents of complex and newly coined and variant technical terminology, company names and so named entity recognition is likely to have an important place in future patent search systems.

## 2.6 Outlook

To summarise the preceding sections of this chapter, the key characteristics of Information Retrieval are:

- Unstructured information, mostly semi-structured data
- No right answers (except for known-item search)
- Separation of indexing and query time processing; offline (crawl/index time) vs. online (query time) processing
- Strong empirical method, reproducibility and evaluation required

What this means for applications such as patent search is the subject of the rest of this book. As outlined in [4], there are indications that iterative search is coming into focus, as newer methodologies such as faceted search or clustering features are becoming more common.

A lot remains to be done. A survey conducted in 2010 [1, 16] compared the features offered by open-source IR systems (from the more academic to industry-strength systems) to the features that are important to patent searchers. While options of the query languages (that depend on the underlying IR models and their extensions) such as wildcards, field operators or proximity search are well-covered, requirements related to the iterative and explorative nature of the search process (which would require greater changes to the whole system) were found to be not covered at all. Functionalities such as combining multiple queries or results, keyword highlighting in the results or grouping the documents by non-explicit meta-data like patent families are missing and have to be implemented outside of the core applications.

## References

1. Azzopardi L, Vanderbauwhede W, Joho H (2010) Search system requirements of patent analysts. In: Proceeding of the 33rd international ACM SIGIR conference on research and development in information retrieval, SIGIR'10. ACM, New York, pp 775–776. URL <http://doi.acm.org/10.1145/1835449.1835610>
2. Baeza-Yates RA, Ribeiro-Neto B (1999) Modern information retrieval. Addison-Wesley Longman Publishing Co, Inc, Harlow
3. Belew RK (2000) Finding out about: a cognitive perspective on search engine technology and the WWW. Cambridge University Press, Cambridge
4. Bonino D, Ciaramella A, Corno F (2010) Review of the state-of-the-art in patent information and forthcoming evolutions in intelligent patent informatics. *World Pat Inf* 32(1):30–38. doi:10.1016/j.wpi.2009.05.008. URL <http://linkinghub.elsevier.com/retrieve/pii/S0172219009000465>
5. Buckley C, Dimmick D, Soboroff I, Voorhees E (2006) Bias and the limits of pooling. In: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'06). ACM, New York, pp 619–620. doi:10.1145/1148170.1148284
6. Buckley C, Dimmick D, Soboroff I, Voorhees E (2007) Bias and the limits of pooling for large collections. *Inf Retr* 10(6):491–508. doi:10.1007/s10791-007-9032-x
7. Buckley C, Robertson S (2008) Relevance feedback track overview. In: Proceedings of the seventeenth text retrieval conference, TREC 2008, Gaithersburg, Maryland, USA, November 18–21, 2008. Special publication 500-277. National Institute of Standards and Technology (NIST), Gaithersburg
8. Bush V (1945) As we may think. *The Atlantic Monthly*. Reprinted in *Life* magazine September 10, 1945

9. Büttcher S, Clarke C, Cormack G (2010) *Information retrieval: Implementing and evaluating search engines*. MIT Press, Cambridge
10. Cleverdon C, Mills J (1963) The testing of index language devices. *Aslib Proc* 15(4):106–130. doi:[10.1108/eb049925](https://doi.org/10.1108/eb049925)
11. Cleverdon CW (1991) The significance of the Cranfield tests on index languages. In: *Proc SIGIR*. ACM Press, New York, pp 3–12
12. Deerwester SC, Dumais ST, Furnas GW, Harshman RA, Landauer TK, Lochbaum KE, Streeter LA (1988) Computer information retrieval using latent semantic structure. URL <http://patft.uspto.gov/netacgi/nph-Parser?patentnumber=4839853>
13. Deerwester SC, Dumais ST, Landauer TK, Furnas GW, Harshman RA (1990) Indexing by latent semantic analysis. *J Am Soc Inf Sci* 41(6):391–407
14. Grossman DA, Frieder O (2004) *Information retrieval: Algorithms and heuristics*, 2nd edn. Springer, Berlin
15. Hunt D, Nguyen L, Rodgers M (2007) *Patent searching: Tools & techniques*. Wiley, New York
16. Joho H, Azzopardi LA, Vanderbauwhede W (2010) A survey of patent users: an analysis of tasks, behavior, search functionality and system requirements. In: *Proceeding of the third symposium on information interaction in context, IiX'10*. ACM, New York, pp 13–24. URL <http://doi.acm.org/10.1145/1840784.1840789>
17. Jones KS, Willett P (eds) (1997) *Readings in information retrieval*. Morgan Kaufmann, San Mateo, pp 167–174
18. Korfhage RR (1997) *Information storage and retrieval*. Wiley, New York
19. Liu TY (2009) Learning to rank for information retrieval. *Found Trends Inf Retr* 3(3):225–331. doi:[10.1561/1500000016](https://doi.org/10.1561/1500000016)
20. Lupu M, Huang J, Zhu J, Tait J (2009) Trec-chem: large scale chemical information retrieval evaluation at trec. *SIGIR Forum* 43:63–70. doi:[10.1145/1670564.1670576](https://doi.org/10.1145/1670564.1670576)
21. Manning CD, Raghavan P, Schütze H (2008) *Introduction to information retrieval*. Cambridge University Press, New York
22. Mooers CE (1950) Coding information retrieval and the rapid selector. *Am Doc* 1(4):225–229
23. Mooers CE (1961) From the point of view of mathematical etc techniques. In: *Towards information retrieval*. Butterworths, Stoneham, pp xvii–xxiii
24. Morville P, Rosenfeld L (2006) *Information architecture for the World Wide Web*. O'Reilly, Sebastopol
25. Robertson S, Spärck Jones K (1994) Simple, proven approaches to text retrieval. Tech Rep UCAM-CL-TR-356, University of Cambridge, Computer Laboratory. URL <http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-356.pdf>
26. Rocchio JJ (1971) Relevance feedback in information retrieval. In: Salton G (ed) *The Smart retrieval system—experiments in automatic document processing*. Prentice-Hall, Englewood Cliffs, pp 313–323
27. Roda G, Tait J, Piroi F, Zenz V (2010) Clef-ip 2009: Retrieval experiments in the intellectual property domain. In: Peters C, Nunzio GD, Kurimo M, Mostefa D, Penas A, Roda G (eds) *Multilingual information access evaluation I*. Text retrieval experiments 10th workshop of the cross-language evaluation forum. LNCS, vol 6241. Springer, Berlin, pp 385–409
28. Rowley J, Farrow J (2000) *Organizing knowledge: An introduction to managing access to information*, 3rd edn. Ashgate Publishing, Farnham
29. Salton G, Wong A, Yang CS (1975) A vector space model for automatic indexing. *Commun ACM* 18:613–620. doi:[10.1145/361219.361220](https://doi.org/10.1145/361219.361220)
30. Schamber L (1994) Relevance and information behavior. *Annu Rev Inf Sci Technol* 29:3–48
31. Schrettinger M (1803) *Versuch eines vollständigen Lehrbuches der Bibliothek-Wissenschaft*. Munich
32. Singhal A (2001) Modern information retrieval: A brief overview. *IEEE Data Eng Bull* 24(4):35–43
33. Turtle HR (1994) Natural language vs. Boolean query evaluation: A comparison of retrieval performance. In: *Proceedings of the 17th annual international ACM-SIGIR conference on*

research and development in information retrieval, Dublin, Ireland, 3–6 July 1994. Special issue of the SIGIR forum. ACM/Springer, New York, pp 212–220

34. Zaragoza H, Craswell N, Taylor M, Saria S, Robertson S (2004) Web and hard tracks. In: Proceedings of the thirteenth text retrieval conference, TREC 2004, Gaithersburg, Maryland, November 16–19, 2004. Special publication 500-261. National Institute of Standards and Technology (NIST), Gaithersburg

## Part II

# Evaluating Patent Retrieval

As was noted in the previous chapter, Information Retrieval as a scientific subject is characterized by a strongly empirical approach, backed up by a rigorous approach to experimental methodology.

A key question for anyone selecting systems to search patents is whether one system or another is better for the purpose of interest. A core value of Information Retrieval as an academic subject is that “improvements” in systems must be rigorously tested, to determine whether they actually deliver better results than previous systems. This has led to the development of well thought through empirical methods in information retrieval, which have influenced commercial practice of well-known search companies, like Google, as well as academic practice in formal international evaluation campaigns like NTCIR, TREC and CLEF.

This part will begin by introducing the basis of IR experimental methodologies. It will then go on to discuss some recent IR evaluation campaigns focused on patent search and conclude with what needs to be done to make these campaigns more relevant to the needs of real patent searchers.

In this part there are four chapters. The first, by Carterette and Voorhees, gives an overview of the whole subject of information retrieval evaluation and experiment including an introduction of key terminology, the standard approaches to evaluation (and its shortcomings) and some pointers to further reading for those interested. The next two chapters cover two major activities (TREC-CHEM and CLEF-IP) to evaluate patent search oriented retrieval systems, which have taken place in the last few years. The last chapter in this part, by Trippe and Ruthven, explores the needs for a framework that would allow formal, fair and repeatable evaluation frameworks to be developed which better reflected the needs and priorities of practical patent searchers than current research evaluation frameworks.

TREC-CHEM and CLEF-IP build significantly on two earlier related activities: a workshop at the SIGIR conference in 2000 and the various NTCIR activities on patent related tasks (see the references within these chapters for further details). It had been hoped to have a chapter reviewing the NTCIR patent related activity in this volume, but that proved impossible in the end.

An important assumption in IR evaluation is that we are trying to measure the effectiveness of the search system (indexer, query processor etc.) INDEPENDENTLY of the data being searched. In practice, for most operational trials (as Trippe and Ruthven call them) of commercial patent search services, one cannot distinguish

the issues in the quality of the search system from issues in the quality of the underlying data feeds. This issue, which goes beyond the scope of this volume, perhaps deserves more thought in the patent search community. On the one hand, the IR scientists need to adapt their evaluation measures and methods, or to create new ones, for the issues specific to this domain. On the other hand, professional patent searchers, as well as commercial vendors, must make their processes more easily subjected to an objective, scientific evaluation.

# Chapter 3

## Overview of Information Retrieval Evaluation

Ben Carterette and Ellen M. Voorhees

**Abstract** An important property of information retrieval (IR) system performance is its *effectiveness* at finding and ranking relevant documents in response to a user query. Research and development in IR requires rapid evaluation of effectiveness in order to test new approaches. This chapter covers the test collections required to evaluate effectiveness as well as traditional and newer measures of effectiveness.

### 3.1 Introduction

Information retrieval systems help users complete search tasks, quite often involving finding a handful of relevant documents among thousands and thousands of pages of text with little structural organization. This is a hard problem: because of the vagaries of natural language and the difficulty of understanding the user's ultimate goal, there is always a good chance that a system given a keyword query will return documents that are not relevant to the user, or that it will fail to find some of the most relevant documents that exist in the collection. Furthermore, users cannot form objective assessments of a system's performance just by working with that system; there are simply too many different factors that influence a user's experience—its user interface, its response time, the user's prior knowledge, etc.—that have nothing to do with the relevance of the results. At the same time, developers of retrieval systems must be able to objectively understand the effects of many different internal factors on the relevance of the end results, as there may be hundreds of features and design decisions that go into building a large search system.

*Evaluation and experimentation* allow developers and researchers to measure and compare aspects of system performance under different conditions as objectively as possible. While many of the properties that affect user experience are important and

---

B. Carterette (✉)  
University of Delaware, Newark, DE 19716, USA  
e-mail: [carteret@cis.udel.edu](mailto:carteret@cis.udel.edu)

E.M. Voorhees  
NIST, Gaithersburg, MD 20879, USA  
e-mail: [Ellen.Voorhees@nist.gov](mailto:Ellen.Voorhees@nist.gov)



measurable (including general interface usability, query response time, and other efficiency issues), the factor that most determines how useful a system is to its users is its *effectiveness*—the overall relevance of results it retrieves in response to a user query. This chapter provides an overview of the process of measuring effectiveness: the data required, the means by which it is collected, and the calculation of the measurements themselves.

Broadly speaking, there are two classes of evaluation: *user-based*, in which actual users interact with a system in a controlled setting, and *system-based*, in which users are essentially simulated by an unchanging set of information needs. User-based evaluations are very valuable, but impractical for many of the day-to-day needs in development and research. Determining the right controls can be quite challenging, and bringing in users and training and monitoring them is time-consuming. It is likely not possible for individual users to distinguish between small effects due to one of many system design decisions, though these effects can become important when scaling up to thousands of users.

System-based evaluations are also called “batch” evaluations, because they involve submitting to a system a batch of pre-fabricated queries derived from a fixed representative set of information needs, then measuring the relevance of the ranked results with no human intervention at any step in the process (apart from setting the batch process running). Because there is no human component, many batch evaluations can be done very quickly, allowing a rapid development cycle. With a careful enough measurement process, the developer can identify very fine differences in effectiveness that may not be noticeable to individual users. This approach has its weaknesses, most notably that it “hides” a lot of information about queries and documents that would be noticed by real users. Nevertheless, it is so valuable that it is the standard approached used for system design and testing.

This chapter focuses on system-based evaluations, as they are currently the primary means by which researchers and developers understand system effectiveness. The Text REtrieval Conference (TREC), organized by researchers at NIST since 1992, performs system-based evaluations [11, 21], as do similar evaluation venues such as NTCIR (NII Test Collections for Information Retrieval, organized by the National Institute of Informatics in Japan), CLEF (the Cross-Language Evaluation Forum organized by the Istituto di Scienza e Tecnologie dell’Informazione), FIRE (the Forum for Information Retrieval Evaluation organized by the Information Retrieval Society of India), and INEX (the INitiative for the Evaluation of XML Retrieval). Readers interested in user-based evaluations are encouraged to read about the evolution of the TREC Interactive Tracks [10] in *TREC: Experimentation and Evaluation in Information Retrieval* [21] and the special issue of the journal *Information Processing & Management* on interactive information retrieval [4].

### 3.1.1 The Cranfield Tests

The original system-based evaluations were the Cranfield tests done in the 1950s and 1960s by Cyril Cleverdon, a librarian and computer scientist in the College of

Aeronautics at Cranfield, UK. Cleverdon identified two broad types of “devices” that affect effectiveness in different ways; he called those that increased the proportion of relevant documents among those retrieved “precision devices” and those that increased the proportion of all relevant documents found “recall devices” [8]. Precision and recall devices could be combined in different ways to vary system behavior in response to user queries; the challenge was measuring the effect of any given combination.

Cleverdon’s idea was simple: rather than run a user study from scratch, he found a group of users who had *already* accomplished a task using some search system, then worked backwards to figure out what steps might lead to the same result. The users were authors of research papers in aeronautics engineering; Cleverdon asked them to describe the research question that inspired the work. He then asked the researchers to rate each of their cited references on a scale of 1–5 for relevance to the research question. Together, the research questions and ratings provided data by which he could simulate a user study: the research questions formed a set of information needs similar to those of the library’s patrons, and the judgments of relevance indicate which articles would be better to retrieve.

This methodology was later adopted by Gerard Salton of Cornell for the evaluation of the highly influential automatic text indexing and search system SMART [15]. As a result, it is now called the “Cranfield paradigm” and has become the *de facto* approach to effectiveness evaluation.

## 3.2 Test Collections

An information retrieval experiment begins with a retrieval task, something that users want to do with an IR system. Examples of tasks include *ad hoc* retrieval (a user wants to find all relevant documents for an arbitrary query), filtering (a user wants to filter the relevant documents from an incoming stream [14]), known-item retrieval (a user wants to find something that they know exists [3]), novel-item retrieval (a user wants to find new relevant documents [12]), and diversity retrieval (different users have different needs for the same query and the system must satisfy them all [7]). The rest of the experimental environment flows from the task.

A *test collection* encapsulates the experimental environment. A test collection is meant to model users with information needs that are particular instances or examples of the task [18, 19]. These information needs are generally treated as if they do not change over time; if they are representative of the needs of users of the system in general, then showing that a system can perform well on them suggests that a system will perform well.

Test collections have three components:

1. a corpus of documents to search;
2. a set of user information needs;
3. judgments of the relevance of information needs to documents in the corpus.

The corpus is the largest part of the test collection, but usually the easiest to obtain, as the existence of a large set of documents to be searched is the *raison d'être* for a retrieval system. Some examples are the newswire corpora assembled for the early TREC conferences, consisting of about one million full-text news articles (the TIPSTER corpora); the 25 million web pages crawled from the .gov domain (the GOV2 corpus); one billion web pages crawled from the general web (ClueWeb09); and the 1.2 million patents from the chemical domain.

### 3.2.1 User Information Needs

The ability of the system to satisfy users' information needs is what we want to measure, so these needs must be carefully constructed not only to reflect the types of things users will do with the system, but also to be able to capture subtle differences in performance between different systems. If the needs are easily satisfied, all systems will appear to be roughly equally good; if the needs are difficult to satisfy, all systems will seem quite poor. Additionally, the information need must be precisely defined so that it is clear to assessors what it means for a document to be relevant to that need. Thus, while it may be simple to take a sample of keyword or Boolean queries from a log, that is usually not sufficient for a test collection. The queries must be fleshed out into full descriptions of the information need, or alternatively the information needs can be developed with certain goals in mind, then the queries derived from that need.

The fully fleshed information need is called a *topic*. Topics usually comprise a keyword query that will be submitted to the retrieval system, a longer description of the information need written in full sentences, and a narrative of what specific types of information should and should not be considered relevant. An example topic developed for the TREC ad hoc task in 1999 is shown in Fig. 3.1.

The number of topics that need to be part of a test collection depends on a host of factors, some of which are quite technical. As a rule of thumb, 50–150 different topics has traditionally been considered sufficient.

### 3.2.2 Relevance Judgments

The relevance judgments tell us which documents are relevant to each of the information needs. As described above, since it is people that will be using the documents, relevance is something that must be determined by people. The system itself can only try to predict relevance; an evaluation determines how good the system is at predicting what will be relevant, and an experiment tells us whether one system is better at it than another.

Once the topics have been finalized, human assessors can start judging documents for relevance. Ideally the person that formulated the topic is also the assessor

```
<top>

<number>425</number>
<title> counterfeiting money </title>

<description>
What counterfeiting of money is being done in modern times?
</description>

<narrative>
Relevant documents must cite actual instances of counterfeiting.
Anti-counterfeiting measures by themselves are not relevant.
</narrative>

</top>
```

**Fig. 3.1** An example topic from the TREC-8 ad hoc retrieval task

judging documents for that topic, although that is not always possible (and sometimes it is useful to have more than one assessor judge the same documents). Assessors read documents, compare them to the topic definition, and say whether they are relevant or not (or possibly how relevant they are).

Exhaustively judging relevance—that is, judging every single document in the corpus to every single topic—is the only way to guarantee that all relevant documents are known. This is often impossible due to time and budget constraints, however. One assessor judging a million documents at a relatively quick rate of 10 per minute would take over ten months of 40-hour weeks to complete just one topic.

Focusing judgment effort on a small portion of the complete corpus can usually provide enough of the relevant documents for most evaluation and experimentation purposes. One simple approach is the *pooling method*: each topic in the collection is submitted to a variety of different retrieval systems, and the top  $N$  ranked documents from all of those systems are pooled for judging [17]. Pooling can be expected to miss some relevant documents [22], but it limits judging to those documents that are least likely to be nonrelevant.

Forming a pool that captures enough relevant documents to be useful requires going to sufficient depth in a set of systems with enough variability to find a diverse set of possibly relevant documents. This means that an adequate pool size depends on the total size of the corpus: as corpora have grown from thousands of documents to millions, pool sizes have grown with them [5]. This, of course, means that pooling eventually becomes too costly itself.

A variety of other methods for deciding which documents to judge have been proposed. Some of these include interactive searching and judging, in which assessors are given a “live” search system with which they interact [9]; statistical sampling, in which documents are sampled from the pool according to a sampling prior probability distribution [2]; adaptive algorithmic approaches, in which an algorithm picks documents to be judged based on dynamic criteria [6, 9]; and citation analy-

sis, in which cited references are assumed to have some degree of relevance [13]. In particular, the statistical sampling approach was specifically designed to produce a good estimate of the number of relevant documents  $R$  in a given pool. Inspired by methods used in polling, it works by defining a sampling distribution over the pool, choosing a subset to judge according to that distribution, then weighting individual relevant documents inversely by their probability of being sampled.

Assessors can actually disagree quite a bit about which documents are relevant. One large-scale experiment suggested that if two assessors judge the same set of documents for the same information need, the relevant documents they find will only overlap by about 40% on average [20]. While this may seem to invalidate the entire idea of evaluating with human judgments, the same work suggests that these differences do not actually affect our ability to compare systems and determine which are the most effective on average. It may be the case that some documents are “obviously relevant”, and that these are the most important for evaluating systems.

### 3.3 Evaluation Measures

Once a test collection has been finalized, at any time someone may submit a query derived from one of its topics to a retrieval system, obtain the ranked list of retrieved documents, and measure the system’s effectiveness using the relevance judgments for that topic. The IR literature is awash with different evaluation measures meant to measure different aspects of retrieval performance; we will focus on a few of the most widely used.

#### 3.3.1 Precision and Recall

Two of the most basic and most important aspects of effectiveness center on the number of relevant documents retrieved:

1. Of the retrieved documents, how many are relevant?
2. Of all relevant documents in the collection, how many are found in the retrieved set?

When cast as proportions, these are, respectively, called *precision* and *recall*.

As an example, suppose we submit the query “counterfeiting money” from the example topic in Fig. 3.1. The system retrieves 10 documents from a corpus of one million; looking at our relevance judgments, we find that these 10 have been judged as follows: *rel, rel, rel, rel, rel, rel, nonrel, nonrel, rel, rel*. There are 162 known relevant documents in the corpus. The precision of these results is  $8/10 = 0.8$  and the recall is  $8/162 \approx 0.05$ .

Precision and recall are quite coarse. Suppose instead of retrieving 10 documents, our system retrieved 50. If 20 of those are relevant, precision and recall are  $20/50 = 0.4$  and  $20/162 \approx 0.12$ , respectively. Though these proportions tell us something

about system performance, there is a lot that they *do not* tell us: How were those relevant documents ranked? Did they appear at positions 1–20, or at positions 30–50, or distributed haphazardly throughout? How many documents will a user have to look at before finding the first relevant document? How much effort in terms of reading documents from the top 50 can the user be expected to put in to find all 20 of those relevant documents?

One solution is to look at precisions and recalls over a series of different rank cutoffs. Rather than look at the entire retrieved set (which will likely be quite large, possibly the entire collection), we pick a rank cutoff, say rank 10, and calculate precision and recall among only the top 10 ranked documents. High precision in the top 10 (*precision@10*) indicates that a user can expect to see a lot of relevant documents near the top, even if the precision of the entire retrieved set is low. High recall in the top 10 (*recall@10*) indicates that there are not many relevant documents remaining to be found after the user has seen the top 10. Trends in precision and recall become apparent over a series of rank cutoffs.

In general, we define precision and recall at rank cutoff  $k$  as

$$\begin{aligned} \textit{precision}@k &= \frac{\# \text{ documents retrieved and relevant up to rank } k}{k}, \\ \textit{recall}@k &= \frac{\# \text{ documents retrieved and relevant up to rank } k}{\# \text{ documents relevant}}. \end{aligned}$$

We note here that if the judgments are not exhaustive, it is possible that the number of relevant documents used to compute recall is an underestimate of the true number. There is also the chance that some of the documents ranked in the top  $k$  will not have been judged. It is convenient to assume those unjudged documents are not relevant; though it may introduce some measurement error, experiments suggest that error is not harmful to the overall evaluation [22].

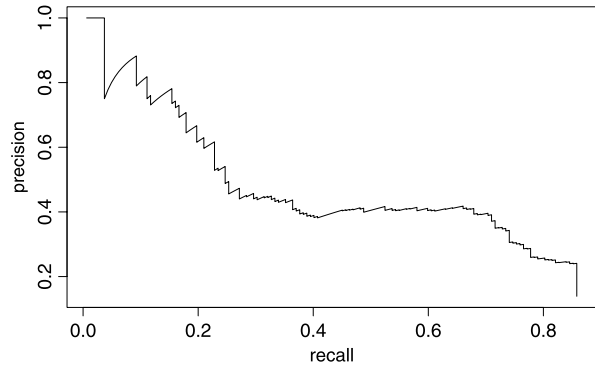
### 3.3.1.1 Precision-Recall Curve

Plotting recall and precision over a series of rank cutoffs produces the *precision-recall curve*. Using raw values of precision and recall at every possible rank cutoff produces a jagged curve like the one shown in Fig. 3.2. This is because recall can never decrease with rank cutoff, while precision increases with every increase in recall and decreases while recall stays constant.

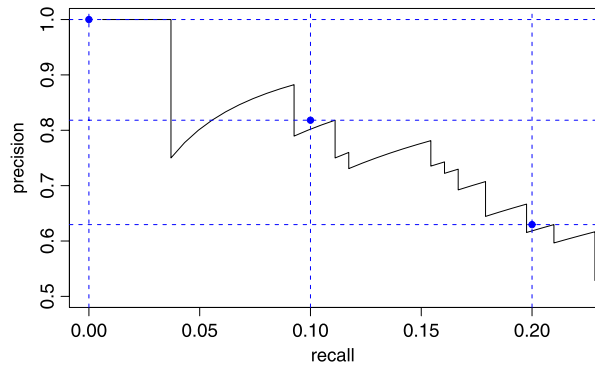
To produce a smoother curve we use a technique called *interpolation*. Interpolated precision is defined by a value of recall rather than by a rank cutoff; specifically, for a given recall level  $r$ , interpolated precision at  $r$  is defined to be the maximum measured precision at any rank cutoff  $k$  at which recall is no less than  $r$ . We formulate this as

$$i\text{-precision}@r = \max_{k \text{ s.t. } \textit{recall}@k \geq r} \textit{precision}@k.$$

**Fig. 3.2** An example of precision-recall curve. There are 162 total relevant documents, so recall increases in increments of  $1/162 \approx 0.006$ . Precision initially trends steadily downward as recall increases from 0 to about 0.25, then holds steady as recall increases from 0.25 to about 0.7, after which it begins to fall again



**Fig. 3.3** Interpolating precision at recall points  $r = 0.0, 0.1, 0.2$  (detail of Fig. 3.2). First we locate point  $r$  on the  $x$ -axis (vertical dashed lines), then find the maximum value of precision after that point (horizontal dashed lines). That value is the interpolated precision at  $r$ , illustrated with the solid points



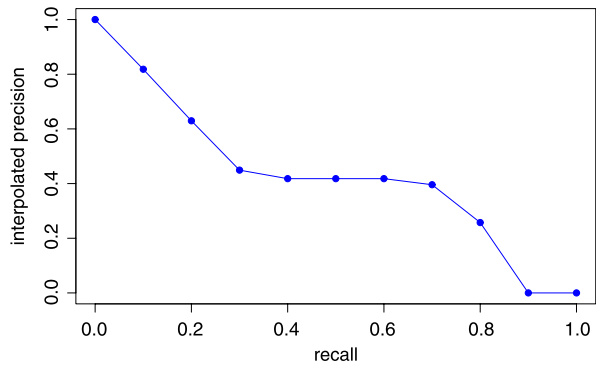
If  $r$  is greater than any recall the system actually achieves, then the interpolated precision is defined to be zero.

Note that  $r$  may not even be an achievable value of recall. In our example Fig. 3.2, there are 162 total relevant documents, which means recall increases by increments of  $1/162 \approx 0.006$ . A recall value of 0.1 is not possible—recall can be  $16/162 \approx 0.099$  or  $17/162 \approx 0.105$ , but not exactly 0.1—yet we can interpolate precision at recall 0.1 nonetheless.

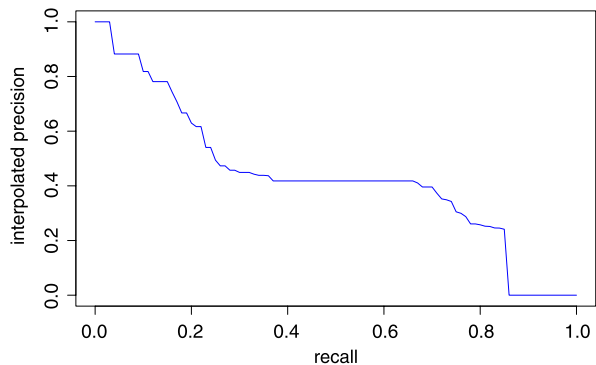
Interpolation is illustrated in Fig. 3.3. Essentially we locate recall point  $r$  on the  $x$ -axis, then find the highest peak of precision that occurs at or after that point. The precision value of that peak becomes the interpolated precision at  $r$ .

Precision is usually interpolated over a set of recall values, then plotted against recall to form a smooth curve such as that shown in Fig. 3.4(a). The *11-point precision-recall curve* is the interpolated curve calculated at the 11 recall values  $\{0.0, 0.1, 0.2, \dots, 1.0\}$ . Smoother curves can be obtained with a finer recall scale; the 101-point curve uses increments of 0.01 (Fig. 3.4(b)).

**Fig. 3.4** The 11-point and 101-point interpolated precision-recall curves computed from Fig. 3.2. The trends in precision are now very clear to the eye



(a) 11-point interpolated precision



(b) 101-point interpolated precision

### 3.3.1.2 Trade-offs Between Precision and Recall

Interpolated precision has the property that it never increases as recall increases. This is because it is defined as the maximum of all precision values after a particular recall point; if  $r_1 < r_2$  and  $\text{i-precision}@r_1$  and  $\text{i-precision}@r_2$  are the maximums of all precisions when recall is greater than  $r_1$  or  $r_2$ , respectively, it follows that  $\text{i-precision}@r_2$  cannot be greater than  $\text{i-precision}@r_1$ .

This has important implications for the design and use of retrieval systems, in that it asks developers and users to choose between higher recall or higher precision. When precision is higher, users will see more relevant documents among the ones they look at, but they will see fewer of the relevant documents that could be found in the collection. When recall is higher, users will have to wade through more nonrelevant documents to find the ones they are looking for, but they will find a larger proportion of all the relevant documents that exist in the collection.

One can ask questions such as “how many relevant documents am I willing to miss out on if it means I save time spent looking at nonrelevant documents?” or “how many nonrelevant documents am I willing to look at to ensure that I find as many of the relevant documents in the collection as possible?” The answers to one of



these questions leads to a point on the precision-recall curve. For instance, using the system illustrated in Fig. 3.3, a user that wants 80% of the documents they look at to be relevant must be willing to accept that they will miss out on 90% of the relevant documents that exist. A user that wants to find 80% of the relevant documents must be willing to accept that 75% of the documents they look at will not be relevant. A developer trying to serve the former user should focus on shifting the left part of the curve up, even if it means a steep drop-off in precision after a certain point (thereby shifting the right part of the curve down). A developer trying to serve the latter user should focus on shifting the right part of the curve up, even if it means losing precision at the lowest recall levels (thereby shifting the left part of the curve down).

### 3.3.1.3 *F*-Measure

It is sometimes useful to look at a single value that summarizes both precision and recall at a certain point. The so-called *F*-measure is the harmonic mean of precision and recall, so defined because the two measures expressed as proportions have the same numerator but different denominators.

$$F@k = \frac{1}{1/\text{precision}@k + 1/\text{recall}@k} = \frac{2 \cdot \text{precision}@k \cdot \text{recall}@k}{\text{precision}@k + \text{recall}@k}.$$

If either precision or recall is zero, *F* is defined to be zero as well. *F* is useful because it allows weighting recall relative to precision in terms of importance. *F* is frequently defined with a weighting parameter  $\beta$  as

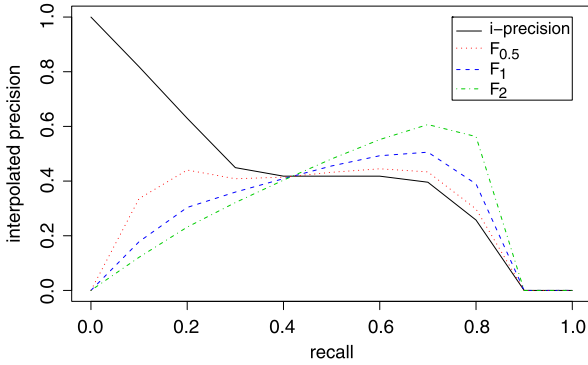
$$F_{\beta}@k = \frac{(1 + \beta^2) \cdot \text{precision}@k \cdot \text{recall}@k}{\beta^2 \cdot \text{precision}@k + \text{recall}@k}.$$

If recall is more important,  $\beta$  can be set higher; if precision is more important,  $\beta$  can be set lower.  $\beta = 2$  weights recall twice as high as precision, while  $\beta = 1/2$  weights precision twice as high as recall.

Like precision and recall, *F* can be computed over a series of rank cutoffs. *F* typically shows an initial increase with rank that gradually levels off and subsequently begins decreasing. Finding the maximum value of *F* over a series of rank cutoff values *k* can indicate an optimal “operating point” at which the tradeoff between precision and recall is best. Figure 3.5 shows  $F_{1/2}$ ,  $F_1$ ,  $F_2$  curves for our example above; the curves suggest an operating point around 70% recall, at which a user would find about 60% of the documents to be nonrelevant.

### 3.3.1.4 Average Precision

Another question we might ask relates to the expected precision at any level of recall, i.e. if we just pick a random point on the curve (corresponding to a rank



**Fig. 3.5** Plotting  $F_\beta$  against recall produces a curve showing how the harmonic mean of precision and recall changes as recall increases. If precision is weighted higher than recall ( $F_{0.5}$  curve), the optimal operating point is around recall 0.2. If recall is weighted higher than precision ( $F_2$  curve), or if precision and recall are equally weighted, the optimal operating point is around recall 0.7

cutoff in the retrieved results), what is the expected proportion of documents that will be relevant? This can be formulated mathematically in various ways, all of which correspond to the total area under a precision-recall curve. Computing the area under the interpolated precision-recall curve results in *average interpolated precision* (AiP) and is equivalent to taking the average of the interpolated precisions calculated over the set of recall points  $S$ .

$$AiP = \frac{1}{|S|} \sum_{r \in S} i\text{-precision}@r.$$

The 11-point curve in Fig. 3.4(a) has  $AiP = 0.4367$ .

Computing the area under the uninterpolated precision-recall curve is just called *average precision* (AP); this is equivalent to taking the average of the precisions calculated at every rank at which a relevant document  $d$  appears.

$$AP = \frac{1}{R} \sum_{d \text{ s.t. } d \text{ relevant}} \text{precision}@rank(d),$$

where  $R$  is the total number of relevant documents in the collection. Note that the sum is calculated over *all* relevant documents, even those that were not retrieved. The precision at an unretrieved document is simply assumed to be zero.

In Fig. 3.2, each increase in the precision-recall curve corresponds to the appearance of a relevant document. The system starts with six relevant documents in a row, so precisions at ranks 1–6 are 1. The next two documents are nonrelevant, but the ninth is relevant. Our computation of average precision would therefore start by summing the six precisions from ranks 1–6 and the precision at rank 9. Continuing in this way over the whole curve, we compute AP to be 0.4253.

Although average interpolated precision was the original means to summarize a curve, it has since been supplanted by AP. Both measures are generally good single-

value summary of the complete precision-recall curve. Since AP is reliable for making fine-grained distinctions between systems, it has become the *de facto* standard for evaluation. However, because it cannot say whether differences occur near the top or at the bottom of the precision-recall curve, it is usually best to consider it in conjunction with other measures and the full curve.

### 3.3.1.5 R-Precision and the Break-Even Point

Another useful summary of the precision-recall curve is the *break-even point*, the point on the curve at which precision and recall are equal. This point corresponds to a rank cutoff value of  $R$ , the total number of relevant documents, and *R-precision* is the name we give to the values of precision and recall at that point. This is the only point on the curve at which it is theoretically possible for both precision and recall to be 100%, and therefore this measure gives a good sense of how far the system is from being perfect.

In practice R-precision and average precision correlate very highly with each other. Like AP, R-precision can be understood as an approximation to the area under the precision-recall curve [1].

### 3.3.1.6 Averaging Precision and Recall Over Topics

The performance of a system on a single topic does not necessarily tell us much about overall system performance: that topic may be “easy” or “hard” in the sense that any system could be expected to do equally well, or it may be unusually easy or hard for a particular system. In either case, a single performance measure gives a distorted sense of system effectiveness. For this reason, the measures above are usually calculated over a set of topics, then averaged to produce a single measure of effectiveness. When averaging a measure over a set of topics, we often attach “mean” to the measure name—e.g., “mean precision@k”, “mean interpolated precision@r”, “mean R-precision”, and “mean average precision” or more succinctly *MAP*.

Even when evaluating over a set of topics, systems may exhibit differences in measured effectiveness that cannot reliably be ascribed to differences in design decisions. This is partly due to the potential for random effects in a relatively small sample of topics; some of them may simply be easier for one system to handle than the other. Whether a measured difference is “real” or not is the question that statistical hypothesis tests attempt to answer. A hypothesis test is a procedure that produces a *p-value* describing the probability of observing a particular set of effectiveness measurements *if* there is no actual difference between the systems. If the *p-value* is low (usually less than 0.05), we conclude that the systems are not equally effective.

One of the most common statistical tests is Student’s *t*-test, which involves computing a test statistic *t* from the mean and variance of the differences between two

systems' measured effectiveness on each topic in a sample of size  $n$ . For instance, if  $AP_{1i}$ ,  $AP_{2i}$ , respectively, indicate the average precisions of systems numbered 1 and 2 on topic  $i$ , then:

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n (AP_{1i} - AP_{2i}), \quad \hat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^n ((AP_{1i} - AP_{2i}) - \hat{\mu})^2,$$

$$t = \frac{\hat{\mu}}{\sqrt{\hat{\sigma}^2/n}}.$$

A value of  $t$  maps to a  $p$ -value which can be found by consulting a  $t$  distribution table. At  $t = 0$ , the  $p$ -value is 0.5, indicating no significant difference between systems; as  $|t|$  increases, the corresponding  $p$ -value goes to zero. Smucker et al. present a fuller description of the  $t$ -test and other common tests along with a comparison of their outcomes [16].

### 3.3.2 Modeling User Effort

One factor of system performance that precision- and recall-based measures do not directly address is the amount of effort a user can be expected to put in while interacting with the system. There are various families of measures that attempt to address this; the most commonly used are the *discounted cumulative gain* (DCG) family and the *rank-biased precision* family. These are families because they depend on a particular model of a user interacting with a ranked list, and different measures arise from defining that model in different ways.

First, one very simple measure along these lines is the reciprocal of the rank at which the first relevant document appears. For example, if the first relevant document appears at rank 2, the reciprocal rank is  $1/2$ . If it is at rank 3, the reciprocal rank is  $1/3$ . The user this measure models has a very strong preference for a relevant document at rank 1 regardless of what the precision-recall curve looks like after that; if there is no relevant document at rank 1, they prefer one at rank 2 but less strongly. When averaged over queries, this measure is called “mean reciprocal rank” (MRR).

#### 3.3.2.1 Discounted Cumulative Gain Family

Discounted cumulative gain (DCG) is defined by a *gain function* and a *discount function*. The gain function reflects the value of a particular relevant document to a user, allowing DCG to take advantage of *grades* of relevance. For instance, relevance judgments may be made on a three-point scale (not relevant, relevant, highly relevant) or a five-point scale (poor, fair, good, excellent, perfect); DCG's gain function can take advantage of these grades by mapping them to numeric values to reflect their utility to a user. Traditional precision and recall can only use binary judgments.

Two typical gain functions are the linear and exponential functions. Linear gain simply assigns incrementally increasing values to each relevance grade, e.g. nonrelevant  $\rightarrow 0$ , relevant  $\rightarrow 1$ , highly relevant  $\rightarrow 2$ . Exponential gain multiplicatively increases values, e.g. poor  $\rightarrow 0$ , fair  $\rightarrow 1$ , good  $\rightarrow 3$ , excellent  $\rightarrow 7$ , perfect  $\rightarrow 15$ . By tuning the gain function, a developer can model users that have varying degrees of preference for different grades of relevance.

The discount function reflects the patience a user has for proceeding down the ranked list. Discounts are assigned to ranks such that discounts never increase with rank. The discount function is usually logarithmic.

Once a gain function  $g$  and a discount function  $d$  have been defined, we can define the discounted gain at any rank as the ratio of the gain of the document at that rank to the discount of that rank:

$$\text{discounted gain}@k = \frac{g(\text{rel}_k)}{d(k)}.$$

DCG@ $k$  is then defined as the sum of the discounted gains from ranks 1 to  $k$ :

$$\text{DCG}@k = \sum_{i=1}^k \frac{g(\text{rel}_i)}{d(i)}.$$

With a linear gain and logarithmic discount, this would be

$$\text{DCG}@k = \sum_{i=1}^k \frac{\text{rel}_i}{\log_2(i+1)}.$$

The range of DCG depends heavily on the relevant documents known for the topic. If there are many highly relevant documents, DCG can be quite high. With a five-point relevance scale, exponential gain, and logarithmic discount, DCG@10 could have a maximum value as high as 68 (if there are 10 “perfect” documents) or a maximum value of only 1 (if there is only one relevant document and it is merely “fair”). This makes averaging DCG over queries somewhat problematic in that the best possible performance varies by topic.

To address this, we can normalize DCG by the maximum achievable DCG (the *ideal DCG*) at the same rank. Ideal DCG is easily found by calculating the DCG of a ranked list that places all the highest-graded documents above all the second-highest-graded documents and so on. Normalized DCG at rank  $k$  (nDCG@ $k$ ) is then computed by dividing DCG@ $k$  by the ideal DCG@ $k$ . nDCG always ranges between 0 and 1 (except when there are no known relevant documents), and is therefore more appropriate for averaging over queries.

### 3.3.2.2 Rank-Biased Precision Family

Rank-biased precision (RBP) models a user starting from the top of the ranked list and deciding whether or not to go on to the next document. The user model consists of a “persistence parameter”  $p$ , defined as the probability that a user goes on to

view the next document. Using this parameter, we can compute the probability that a user will view exactly the first  $k$  documents as  $p^{k-1}(1-p)$ , i.e. the probability that a user moves from first to second, times the probability of moving from second to third, and so on down to the  $k$ th ranked document, finally multiplying by  $(1-p)$  as the probability that the user does *not* move on to the  $k+1$ st ranked document. Multiplying the probability of viewing the first  $k$  documents by the proportion relevant in those  $k$  provides a user-based view of precision; averaging over all  $k$  in effect gives an expectation of the performance a user with a particular value of  $p$  will experience.

If  $rel_i$  is the relevance of document  $i$  in the ranking, RBP can be expressed as

$$RBP = (1-p) \sum_{i=1}^n rel_i p^{i-1}.$$

RBP effectively uses geometric discounting, with each rank given a weight of  $p$  times the previous. If, for example,  $p$  is set to 0.8, rank 1 has weight  $0.8^0 = 1$ , rank 2 has weight  $0.8^1 = 0.8$ , rank 3 has weight  $0.8^2 = 0.64$ , and so on. As the rank increases, the weight decreases, getting closer and closer to zero. In this case, at rank 21 the weight is  $0.8^{20} \approx 0.01$ , which is sufficiently low that documents at that rank have virtually no effect on the final value of the measure. RBP's discounting is much stricter than DCG's for typical choices of  $p$  and discount function; for comparison, the rank at which DCG's logarithmic discount would be 0.01 is on the order of  $10^{30}$ .

### 3.4 Conclusion

Effectiveness evaluation is an important aspect of research and design of information retrieval systems. Much research has been done on the topic, and more continues to appear every year. The issue of cost-effective relevance judging and evaluation remains important. Interest in devising user models for evaluations that go beyond individual, independent document relevance has recently increased; ongoing work in *novelty and diversity* is investigating the tradeoff between the relevance of documents and the redundancy of relevant information within the documents, while work on *query sessions* looks at effectiveness over a sequence of user interactions with a system. Such problems require new types of test collections, new types of relevance judgments, and new evaluation measures, representing new frontiers in effectiveness measurement research.

### References

1. Aslam J, Yilmaz E, Pavlu V (2005) A geometric analysis and interpretation of R-precision. In: Proceedings of CIKM, pp 664–671
2. Aslam JA, Pavlu V (2008) A practical sampling strategy for efficient retrieval evaluation. Northeastern University tech report

3. Beitzel SM, Jensen EC, Chowdhury A, Grossman D, Frieder O (2003) Using manually-built web directories for automatic evaluation of known-item retrieval. In: Callan J, Hawking D, Smeaton A (eds) SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval, July 28–August 1. ACM, New York, pp 373–374
4. Borlund P, Ruthven I (eds) (2008) *Inf Process Manag* 44(1). Special issue
5. Buckley C, Dimmick D, Soboroff I, Voorhees E (2006) Bias and the limits of pooling. In: Dumais ST, Efthimiadis EN, Hawking D, Järvelin K (eds) SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, August 6–August 11. ACM, New York, pp 619–620
6. Carterette B, Allan J, Sitaraman RK (2006) Minimal test collections for retrieval evaluation. In: Dumais ST, Efthimiadis EN, Hawking D, Järvelin K (eds) SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, August 6–August 11. ACM, New York, pp 268–275
7. Clarke CLA, Craswell N, Soboroff I (2009) Overview of the TREC 2009 Web track. In: Voorhees EM, Buckland LP (eds) Proceedings of the 18th text retrieval conference (TREC 2009), Nov 2009. NIST, Gaithersburg
8. Cleverdon CW, Mills J (1997) The testing of index language devices. In: Spärck Jones K, Willett P (eds) Readings in information retrieval. Morgan Kaufmann, San Francisco, pp 98–110
9. Cormack GV, Palmer CR, Clarke CL (1998) Efficient construction of large test collections. In: Croft WB, Moffat A, van Rijsbergen CJ, Wilkinson R, Zobel J (eds) SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, Aug 24–28. ACM, New York, pp 282–289
10. Dumais ST, Belkin NJ (2005) The TREC interactive tracks: Putting the user into search. In: Voorhees EM, Harman DK (eds) TREC: Experiment and evaluation in information retrieval. MIT Press, Cambridge, pp 123–152
11. Harman D (1997) The TREC conferences. In: Spärck Jones K, Willett P (eds) Readings in information retrieval. Morgan Kaufmann, San Francisco, pp 247–256
12. Harman D (2002) Overview of the TREC 2002 novelty track. In: Voorhees E (ed) Proceedings of the 11th text retrieval conference (TREC 2002), Nov 2002. NIST, Gaithersburg, pp 46–55
13. Lupu M, Piroi F, Huang X, Zhu J, Tait J (2009) Overview of the TREC 2009 chemical IR track. In: Voorhees EM, Buckland LP (eds) Proceedings of the 18th text retrieval conference (TREC 2009), Nov 2009. NIST, Gaithersburg
14. Robertson S, Hull DA (2000) The TREC-9 filtering track final report. In: Voorhees EM, Harman DK (eds) Proceedings of the 9th text retrieval conference (TREC-9) Nov 2000. NIST, Gaithersburg
15. Salton G, Lesk ME (1997) Computer evaluation of indexing and text processing. In: Spärck Jones K, Willett P (eds) Readings in information retrieval. Morgan Kaufmann, San Francisco, pp 60–84
16. Smucker M, Allan J, Carterette B (2007) A comparison of statistical significance tests for information retrieval evaluation. In: Proceedings of CIKM, pp 623–632
17. Spärck Jones K, van Rijsbergen CJ (1976) Information retrieval test collections. *J Doc* 32(1):59–75
18. Tague J (1981) The pragmatics of information retrieval evaluation. In: Spärck Jones K (ed) Information retrieval experiment. Butterworth, London, pp 59–102
19. Tague-Sutcliffe J (1997) The pragmatics of information retrieval evaluation revisited. In: Spärck Jones K, Willett P (eds) Readings in information retrieval. Morgan Kaufmann, San Francisco, pp 205–216
20. Voorhees E (1998) Variations in relevance judgments and the measurement of retrieval effectiveness. In: Croft WB, Moffat A, van Rijsbergen CJ, Wilkinson R, Zobel J (eds) SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, Aug 24–28. ACM, New York, pp 315–323

21. Voorhees EM, Harman DK (eds) (2005) TREC: Experiment and evaluation in information retrieval. MIT Press, Cambridge
22. Zobel J (1998) How reliable are the results of large-scale information retrieval experiments? In: Croft WB, Moffat A, van Rijsbergen CJ, Wilkinson R, Zobel J (eds) SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, Aug 24–28. ACM, New York, pp 307–314



# Chapter 4

## Evaluating Information Retrieval in the Intellectual Property Domain: The CLEF–IP Campaign

Florina Piroi and Veronika Zenz

**Abstract** The CLEF–IP track ran for the first time within the CLEF 2009 campaign. The purpose of the track was twofold: (a) to encourage and facilitate research in the area of patent retrieval by providing a large clean data set for experimentation; (b) to create a large test collection of patents in the three main European languages for the evaluation of cross-lingual information access. The track focused on the task of prior art search, to which a second task was added in 2010, the patent classification task. The participating teams deployed a variety of Information Retrieval techniques, adapted or custom-made, to tackle with this specific domain and tasks. This chapter reports on activities undertaken to provide a set of topics for the two tasks, to extract the relevance assessments for the provided topics, and on evaluating the effectiveness of the employed retrieval methods.

### 4.1 Introduction

The Cross Language Evaluation Forum CLEF [2] originally arose from work on Cross Lingual Information Retrieval in the US National Institute of Standards and Technology Text Retrieval Conference TREC [15] but has been run separately since 2000. Every year, a number of tasks on both cross-lingual information retrieval (CLIR) and monolingual information retrieval in non-English languages have been run. In 2008 the Information Retrieval Facility (IRF) and Matrixware Information Services GmbH obtained the agreement to run a track which allowed groups to assess their systems on a large collection of patent documents containing a mixture of English, French and German documents derived from European Patent Office data. This became known as the CLEF–IP track, which investigates IR techniques in the Intellectual Property (IP) domain.

---

F. Piroi (✉)

Information Retrieval Facility, Vienna, Austria  
e-mail: [f.piroi@ir-facility.org](mailto:f.piroi@ir-facility.org)

V. Zenz

max.recall information systems, Vienna, Austria  
e-mail: [v.zenz@max-recall.com](mailto:v.zenz@max-recall.com)

One main requirement for a patent to be granted is that the invention it describes be novel: that is, there should be no earlier patent or other publication describing the invention. The novelty breaking document can be published anywhere in any language. Hence when a person undertakes a search, for example to determine whether an idea is potentially patentable, or to try to prove a patent should not have been granted (a so-called opposition search), the search is inherently cross-lingual.

Although there is important previous academic research work on patent retrieval (see for example the ACM SIGIR 2000 Workshop [9] or more recently the NTCIR workshop series [4]), there was little work involving non-English European Languages and participation by European groups was low. CLEF-IP grew out of desire to promote such European research work and also to encourage academic use of a large clean collection of patents being made available to researchers.

This chapter presents the first two CLEF-IP evaluation campaigns, which ran in 2009 [13], as a track of the Cross Language Evaluation Forum CLEF, and in 2010 [11], as a benchmarking activity of the Conference on Multilingual and Multimodal Information Access Evaluation 2010 [1]. Although it would be unreasonable to pretend the work is beyond criticism it does represent a significant step forward for both IR community and patent searchers.

## 4.2 The CLEF-IP Collection

### 4.2.1 *On Patents and Patent Documents*

The CLEF-IP collection contains patents, physically stored as a collection of patent documents. A patent document may be an application document, a search report, or a granted patent document. We describe in the following some of the key terms and steps in a patent's life-cycle. Most of the notions we explain in this section have been defined or commented on in the introductory chapter of this book. We restate their meaning in this subsection mainly for keeping the chapter self-contained and to make these legal notions more understandable to the track's participants, who are part of the IR research community.<sup>1</sup>

A patent is a set of exclusive legal rights for the use and exploitation of an invention in exchange for its public disclosure. The exclusive rights are given by a governing authority and are limited in time. The requirements for granting patents vary widely among patent offices, but a common first step is to file a patent application request with a patent office. For this, the applicant must supply a written specification of the invention—also called an *application document*—where the background of the invention, a description of the invention, and a set of claims which define the scope of protection, should the patent be granted, are given. The *application date*, or *filing date* of a patent refers to the date when the patent application was filed.

---

<sup>1</sup>It is our direct experience that these explanations helped IR researchers the most in understanding the relationships between the different kinds of patent documents constituting a patent.

It is important to note that various parts of a patent application use different types of language, and are usually written by different persons. The invention abstract and description sections use a natural language and are written principally by the inventor, while the claims section use a legal type of language following certain legal rules and is written by patent attorneys.

In order to be granted, a patent application is examined by professionals who will analyze whether it meets certain patentability criteria and whether the application complies with the relevant patent law. The most important patentability criteria are *novelty*, *inventiveness*, and *practicality*. Of relevance to the CLEF-IP benchmarking activity is the novelty criteria. A patent application satisfies the novelty requirement if no earlier patent or other kind of publication, regardless of the publication language, describing (parts of) the invention can be found in a reasonable amount of time. Such a search for novelty-relevant documents is called a *prior art search*. Results of a prior art search are recorded in a *search report*, and are a basis for further communication with the applicant which may result in modifications of the patent specifications before the patent is granted. The relevant documents listed in a search report of a patent are referred to as *patent citations*. Usually, the search report and the application document are published within 18 months from the application date.

When a patent application is found to meet all the necessary legal and patentability requirements, a decision to grant the patent is made and, after further fees and procedural steps, the granted patent is published. An important procedural step at the EPO is that a translation of the claims in all three official EPO languages (English, German, French) is provided [3].

Patent documents generated at the different stages of the patent's life-cycle are identified by a country code (denoting the patent office analyzing/granting the patent), a unique numeric identifier, and by a kind code together with a version number.<sup>2</sup> In the case of EPO the "A" in the kind code denote a patent document published in the application phase (application document, search report, additional search report, etc.), the "B" kind code marks a granted patent document.<sup>3</sup>

It is possible to file a patent application at more than one patent office. When the same invention is granted a patent by different patent offices, the two patents are said to belong to the same *patent family*. (The notion of *patent family* is a more extensive one than stated here, we direct the reader to Chap. 1 in this book for a more comprehensive description.)

An important tool in organizing the large amount of patent data which patent offices regulate is the *classification system*. A patent classification system 'sorts' the patents according to the technical area they belong to, and it is a basis for a quick investigation of the state of the art in a field.<sup>4</sup> There are several patent classification systems—most of them built hierarchically—the most used ones being the

---

<sup>2</sup>For EP patents, documents at different stages have the same numeric identifier. For other patent offices this is not always the case. For example, the patent document US-6689545-B2 represents a US granted patent with its application document publication number US-2003011722-A1.

<sup>3</sup>For a complete list of kind codes used by various patent offices see <http://tinyurl.com/EPO-kindcodes>.

<sup>4</sup>See <http://www.wipo.int/classifications/ipc/en/>.

International Patent Classification system (IPC), which is used by more than 100 patent offices. Other classification systems are the European Classification System (ECLA), the US Classification System, the Japanese F-term classification system. Finally, we mention that a patent may be tagged with more than one classification codes.

#### 4.2.2 Documents in the Data Collection

CLEF-IP is a large scale evaluation campaign, both in terms of the number of topics and in terms of the size of the document collection. The collection corpus contains patent documents published by the EPO: 1.9 million patent documents with publication dates between years 1985 and 2000 in the 2009 campaign, and 2.6 million patent documents with publication dates up to year 2002 in the 2010 campaign.

The patent documents are provided as XML files conforming to one same Document Type Definition (DTD) and are part of the MAREC data corpus [16].<sup>5</sup> All documents in the CLEF-IP collection contain the following main XML fields: bibliographic data (containing the invention title, filing and priority dates, classification codes, inventor names, etc.), abstract, description, and claims. Not all documents actually have content in these fields. This happens because certain EPO patent applications are internationally filed under the Patent Cooperation Treaty (PCT<sup>6</sup>) in which case, the EPO does not republish the whole patent application, but only a bibliographic entry which refers to the original application.

So far, the campaign involved only the textual content of patents. For the next evaluation cycles we aim to also incorporate images into the evaluation setup.

The collection corpus was delivered to the participants “as is”, without merging the documents related to the same patent into one document. Each patent is identified by a unique patent number—a string starting with “EP” and followed by 7 digits. Corresponding to each patent is a directory containing the patent documents related to that patent. The layout is `nnnnnn/nn/nn/nn/* .xml`, with `n` standing for a digit. For example, to patent EP 0981201 corresponds the directory `000000/98/12/01` which contains the following files: `EP-0981201-A2.xml`, `EP-0981201-A3.xml`, and `EP-0981201-B1.xml`.

All patents in the collection have content in one of the three official EPO languages English, German and French. Depending on the stage a patent is in—application phase, granting phase—the patent document will contain text sections only in one language (applications) or in all three of them (grants). For example, an application document may contain the abstract and claims in German only, while the granted patent document contains additionally the claims also in English and

---

<sup>5</sup>Although the MAREC collection was created after the first CLEF-IP campaign was set up in 2009, the documents in the CLEF-IP’09 corpus are included in the MAREC collection, and use the same DTD.

<sup>6</sup><http://www.wipo.int/pct/en/>.

French. In the XML file, this is reflected in the multiple occurrence of the claims field, with different language attributes, and textual content in the corresponding language. The distribution of patents over languages is uneven, with a dominance of English documents. In the first campaign year 69% of the patent documents in the collection have their main language tagged as English, 23% German and 7% French. The language distribution is similar for the collection distributed in second campaign year as well.

### 4.2.3 Tasks and Topics

In both of the 2009 and 2010 CLEF-IP evaluation campaigns the focus has been set on finding prior art for a given patent. Participants to the campaign were asked to return all patents in the collection which constituted prior art for the given topic patents. Participants could choose among four different topic sets of sizes ranging from 500 to 10,000 in 2009, and two topic sets of sizes 500 and 2,000 in 2010. The language used for retrieving documents was not restricted to any of the three official EPO languages. In the first CLEF-IP campaign three further topic sets (one for English, one for German and one for French) dedicated to cross-lingual search were also proposed. Topics in these additional sets had content only in the respective EPO language.

In addition to the prior art task (PAC), the second CLEF-IP campaign proposed a second kind of task: patent classification (CLS). Participants to this task were asked to classify the given patent documents according to the IPC system, up to the subclass level (recall that the IPC system is a hierarchically built classification system, the levels being sections, classes, subclasses, main groups, subgroups). The set of topics in the classification task contained 2,000 patent documents different from the ones used in the prior art task. Since a patent can be tagged with more than one IPC code, the Classification task organized in 2010 is clearly a multi-classification problem.

Topics for the proposed tasks were selected out of a *topic pool* which contained a different part of the MAREC data corpus from the corpus made available to the participants. The split of the set of EP patents available in MAREC into data corpus and topic pool was done as suggested in [5]. The topic pool used in 2009 contained over 0.7 million patent documents, published between years 2001 and 2006. In 2010 the topic pool contained 0.8 million patent documents published between years 2002 and 2009.

Several restrictions were applied when selecting the topics for the various tasks. The most important one is that the documents selected as potential topics must have content in the various XML fields of their digital representation, especially in the claims field. In addition to this restriction, for the prior art tasks, we also required that the potential topic documents have recorded at least 3 citations in their search reports. In 2009, out of the topic pool documents fulfilling the selection requirements, a number of 10,000 topics for the prior art task were composed from the granted

patent document (kind B1) to which the missing XML fields were added from the other available patent documents for the respective patent. In 2010, out of the patent documents in the topic pool that fulfilled these conditions we selected a number of 2,000 application documents (kind A) for the prior art task. Another 2,000 application documents were selected for the patent classification task, the number of citations restriction being, however, omitted, as it is not relevant for a classification task based on textual content only. The citation information was removed from the documents that were released as PAC topics. In the same spirit, the classification information was removed from the released CLS topics.

Participants that did not have computing power to process all the proposed topics were allowed to submit retrieval results for subsets of the largest topic set made available. In 2009 three such subsets were available, with 500, 1,000, and 5,000 topics. In 2010 only one such subset was made available, with 500 topics.

The drastic decrease in the number of topics released (10,000 vs. 2,000) is motivated by our experiments with the submitted data in 2009. That is, the order of the systems did not change when running evaluations on smaller sets compared to the largest set [13].

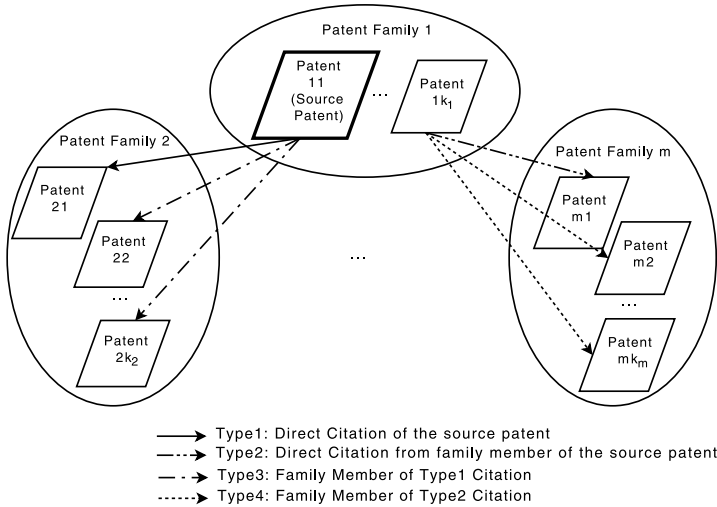
#### ***4.2.4 Relevance Assessments***

A common challenge in IR evaluation is the creation of ground truth data against which to evaluate retrieval systems. The common procedure of pooling and manual assessment is labor-intensive, and, as a further difficulty, voluntary assessors are difficult to find, especially when expert knowledge is required as is the case of the patent field. Researchers in the field of patents and prior art search, however, are in the lucky position of already having partial ground truth at hand. These are the patent citations that are recorded in the search reports attached to patents. As these search reports are publicly available—and also part of the MAREC data collection—we were able to automatize the extraction of relevance judgements for the track's topics.

A general method for generating relevance assessments from patent citations is described in [5]. This idea had already been exploited at the NTCIR workshop series [9]. Further discussions within the 1st IRF Symposium in 2007 led to a clearer formalization of the method.

For both the 2009 and 2010 CLEF-IP campaigns we used an extended list of citations that includes not only patents recorded in the patent's search reports, but also those in the search reports of the family members of the topic patent, as well as the family members of the cited patents. By means of patent families we were able to increase the number of citations by a factor of seven. Figure 4.1 illustrates the process of gathering direct and extended citations.

In the process of gathering citations, patents from ~70 different patent offices (including USPTO, JPO, etc.) were considered. Out of the resulting lists of citations all non-EPO patents were discarded as they were not present in the target data set and thus not relevant to our track.



**Fig. 4.1** Patent citation extension used in CLEF-IP

In using the citations in the published search reports, it is important to know the following.

- Citations have different degrees of relevancy: some patent offices (e.g. USPTO) require applicants to disclose all known relevant publications when applying for a patent. This often leads to applicants listing a large number of prior art patents, not necessarily all highly relevant. Where a patent citation comes from can be spotted easily by the label attached to the citations: APP as coming from the patent applicant, SEA, EXA as coming from patent examiners, OPP as a citation found during an opposition procedure, etc. Patent experts advise to chose topic patents with less than 30 citations coming from the applicant.
- Citation language may differ from the patent application's own publication language: During a novelty kind of examination, especially in the European Union, patent experts must and usually do inspect prior art documents in other (European) languages than the language of the application document. When relevant, these documents are stored into the search report of the patent application.
- The citation lists are incomplete: the nature of the search is such that it often stops when it finds one or only a few documents that are highly relevant for the patent. The Guidelines for examination in the EPO [6] prescribe that if an examination search results in several documents of equal relevance, the search report should normally contain no more than one of them. This means that we have incomplete recall bases, which must be taken into account when interpreting the evaluation results presented here.

Obtaining the assessments for the Classification task required less effort than obtaining those for the Prior Art task. We have used the IPC codes recorded in the bibliographic data fields of the patent documents, which were extracted automatically from the documents chosen as Classification topics.

## 4.3 Submissions

For all CLEF-IP tasks, a *submission* (or *run*) consisted of a single text file with at most 1,000 answers per topic. The format of the submissions followed the standard format used for the TREC submissions, which is a list of tuples containing at least the *topic identifier*, the retrieved *answer*, the *rank* of the retrieved answer, and the *score* given by the retrieval system to the retrieved answer. Table 4.1 shows a list of participating groups and number of runs submitted. The numbers in the parentheses represent the number of runs submitted to the optional language tasks available in the 2009 campaign (English, German, French). In the first campaign year, the runs ranged over all topic sizes, and it was often the case that a participant submitting a run for the largest set, submitted the same run for the other, smaller, topic sets, by restricting the set of topics in the largest run to the ones in the smaller topic sets. This accounts for the large number of submissions in 2009. Considering this fact, the actual number of unique runs with regard to the retrieval method involved is on a par with the number of runs submitted to the prior art task in 2010.

### 4.3.1 Submission Systems

Evaluation campaigns have used a range of track management systems ranging from simple file uploading systems to full-fledged, web-based systems. Almost all, however, are custom tailored for the type of campaign and tasks proposed by the track organizers.

Similarly, the submission system used in CLEF-IP 2009 was custom-built on the open source document management system Alfresco<sup>7</sup> and the web interface Docasu.<sup>8</sup> The system provided an easy-to-use web front-end, which allowed participants to upload and download files. The system offered version control as well as a number of syntactical correctness tests triggered by file submission. The validation tests showed the participants a detailed description of the problematic content of their submissions, most format errors being, therefore, detected automatically and corrected by the participants themselves. Further errors, like finding duplicates in the list of retrieved documents, were done manually after the submission closure. Lack of resources in 2010 stopped us from updating the submission system to be re-used, therefore participants sent us their runs using an ftp server we temporarily made available to them. All validations and checks were done manually after submission closure. Still, the number of corrections that needed to be carried out on the run files was low.

---

<sup>7</sup><http://www.alfresco.com/>.

<sup>8</sup><http://docasu.sourceforge.net/>.



**Table 4.1** List of participants and number of runs submitted.

Institution	2009		2010		
	ID	PAC Runs	ID	PAC Runs	CLS Runs
BiTeM, Service of Medical Informatics, Geneva University Hospitals	CH	hcuge	4 (3)	bitem	7 2
Glasgow Univ.–IR Group Keith	UK	clefip–ug	5		
Geneva University, Centre Universitaire d’Informatique SimpleShift	CH	clefip–unige	5	ssft	8
Centrum Wiskunde & Informatica–Interactive Information Access Spinqe	NL	cwi	4	spq	1 1
Dublin City Univ., School of Computing	IE	clefip–dcu	3	dcu	3
Hildesheim University, Information Science	DE	Hildesheim	1	hild	4
Humboldt Univ., Dept. of German Language and Linguistics and INRIA	DE FR	humb	4 (3)	humb	1 1
LCI–Institut National des Sciences Appliquées de Lyon	FR			insa	5
Industrial Property Documentation Department, JSI Jouve	FR			jve	3
Technical Univ. Valencia, Natural Language Engineering	ES	NLEL	1		
Radboud University Nijmegen	NL	clefip–run	2 (1)	run	2 2
Technical Univ. Darmstadt, Dept. of CS, Ubiquitous Knowledge Processing Lab	DE	TUD	16 (12)		
Al. I. Cuza University of Iași–Natural Language Processing	RO	UAIC	1 (1)	uaic	1
Information Retrieval Group, Universitas Indonesia	ID			ui	3
UNED–E.T.S.I. Informatica, Dpto. Lenguajes y Sistemas Informaticos	ES			uned	8
Univ. Neuchatel–Computer Science	CH	UniNE	8		
Santiago de Compostela Univ., Dept. Electronica y Computacion	ES	uscom	8		
University of Tampere–Info Studies & Interactive Media and Swedish Institute of Computer Science	FI SE	UTASICS	8		
<b>Total:</b>			<b>70 (20)</b>	<b>25</b>	<b>27</b>

### 4.3.2 Short Summary of the Submissions

CLEF-IP presented several challenges to the campaign participants:

- a new retrieval domain (patents) and task (prior art);
- a relatively large-sized collection;
- the language used in patents: documents in the collection contain not only natural style English, German or French text but also patent-specific language, which, in the case of patent claims, has a complex syntactic structure [14];
- topic representations: in most tracks a topic consists of few selected query words or a specific question; In CLEF-IP a topic consists of a whole patent document.

Therefore, to have an overview of the techniques used at CLEF-IP, we looked at how participants approached indexing of the target data, how queries were generated and retrieval results ranked, how the different language content was exploited, as well as how the patent-specific (meta)data were utilized. Details about the retrieval methods and systems participants have used to answer the CLEF-IP challenges are given in Tables 4.2, 4.3, and 4.4. We give here only a summary of these systems and methods, and direct the reader to the CLEF 2009 post proceedings [10] and CLEF 2010 working notes, available on-line on the conference website [1].

The big majority of the participants in both of the CLEF-IP tracks have used off-the-shelf retrieval engines like Indri/Lemur or Terrier (in the PAC task), and k-NN, SVM or Winnow-like classifiers (in the CLS task), choosing to tune these systems in the hope to obtain good results. This includes selecting certain file parts to index, building separate indices per language, or boosting query terms extracted from certain parts of the topic files. Submissions to the Classification task were obtained either using text classifiers only, or by text-retrieval systems (as in the PAC task) giving the IPC codes as results, or by combining classification and text retrieval.

Given that each patent document could contain fields in up to three languages, some participants chose to build separate indices per language, while others generated one mixed-language index or used text fields only in one language discarding information given in the other languages. The granularity of the index varied too, as some participants chose to concatenate all text fields into one index, while others indexed different fields separately. In addition several special indices like phrase or passage indices, concept indices and IPC indices were used. Table 4.2 in the Appendix details which fields were used in creating the indices for the PAC task (columns 2–6), and the systems that were used to create the indices (column 7), as well as other notes of importance to the index creation phase (column 8). Table 4.4 details which document fields were used for training and as input to the classifiers (column 2), if any pre-processing was done (column 3), and which classification/ranking methods were put to use.

As CLEF-IP topics are whole patent documents (with thousands of words), many participants found it necessary to apply some kind of term selection in order to limit the number of terms in the query. Methods for term selection based on term weighting are shown in column 7 of Table 4.3. Columns 2–6 mark which XML fields of the patent documents in the collection were used by the query term selection

methods. The bibliographic data that was exploited the most is the IPC information which was used either as post-processing filter or as part of the query. The patent citation information stored in the document set of the collection was exploited less in the first year, with more groups using this metadata in the second CLEF-IP year. Other very patent-specific information, like priority, applicant, inventor information was only rarely used.

Concerning cross-linguality, not all participants focused on the multilingual nature of the CLEF-IP document collection. In most cases they used only data in one specific language or implemented several monolingual retrieval systems and merged their results. In 2009 only few participating groups made use of machine translation (see column 10 of Table 4.3 in the Appendix). When certain fields in the topic documents were missing for one of the three EPO languages, one group used Google Translate to obtain query terms in the missing language. They report that using the Google translation engine actually deteriorated their results. The best performing group used cross-lingual concept tagging on the documents in the collection in order to create a multilingual terminological database which was used then to create multilingual queries. The situation did not change in 2010, where Google Translate was still the tool of choice to invoke in machine translation, but most of the participants chose to ignore the multi-lingual attributes of the collection.

## 4.4 Measurements and Results

To evaluate the retrieval efficiency of the submitted experiments to the CLEF-IP tasks we have chosen the most commonly used metrics in IR effectiveness evaluation. For the prior art tasks these included Precision and Recall at various cut-offs, MAP and NDCG [7], in the first year, to which we have additionally computed the PRES score [8] in the second year. The measures for the classification task included Precision, Recall,  $F_1$  at various cut-offs and MAP.

In the track's first year—where only the prior art task was organized—the measures were computed with SOIRE [3], and double-checked against `trec-eval`<sup>9</sup> the latter being commonly used in the TREC evaluation campaigns. At that time, there was no available implementation of the NDCG measure, so we used an own implementation that was not using a cumulated gain factor as described in [7]. In the second year we have used `trec-eval` version 9.0 for all the measures, with the exception of PRES where we used the implementation provided by the score's authors.

MAP, Recall@100 and Precision@100 for the small set of topics of the two campaigns are shown in Figs. 4.2 and 4.3. The score results for the NDCG and of the (experimental) PRES measure for the prior art runs submitted in 2010 are shown in Fig. 4.4. Figure 4.5 shows a plot of the computed measures for the classification task in the CLEF-IP 2010 campaign. Detailed reports on the evaluation activities done during the campaigns presented in this chapter can be found in the technical reports [13, 14] and [12].

---

<sup>9</sup>`trec-eval` version 8.0 [http://trec.nist.gov/trec\\_eval](http://trec.nist.gov/trec_eval).

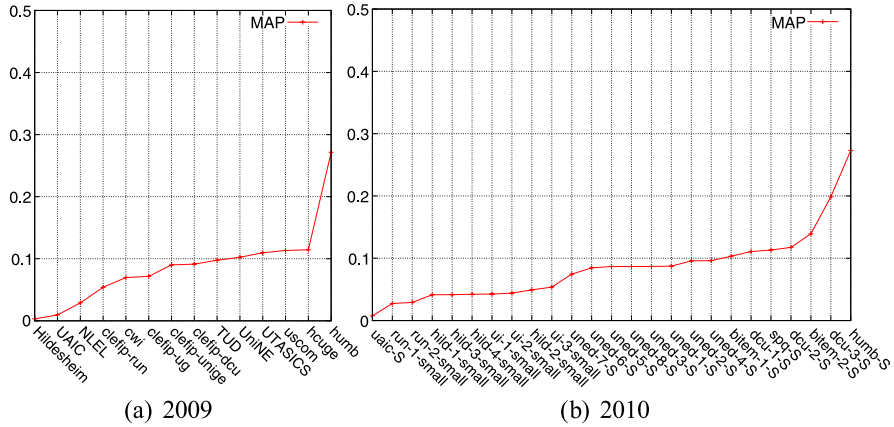


Fig. 4.2 MAP measure values for the prior art tasks

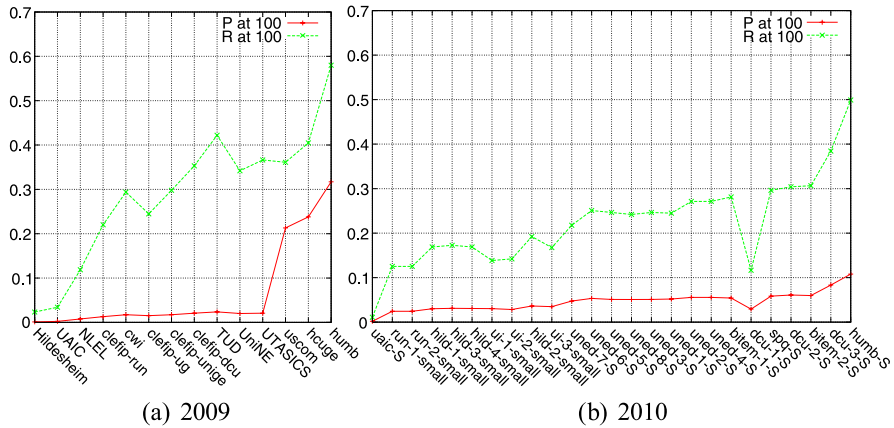


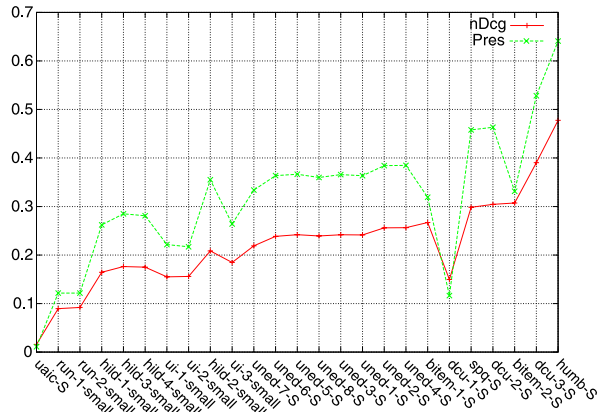
Fig. 4.3 Precision and Recall measure values at cutoff 100, prior art tasks

### 4.5 Closing Words

At the end of the first two CLEF-IP evaluation campaigns, it is clear to us that successful information retrieval in the patent domain involves at least well-thought adjustments to the currently used retrieval and text mining systems. Even so, retrieval results do not come very close to the expectations of patent experts. One reason for this is that transferring the know-how an IP professional has to the IR research community is not a highway as barrier free as we would like.

The CLEF-IP campaigns here described are focused on text-oriented information retrieval. Other than that, there are further aspects of the patent domain information retrieval that can and should be investigated. One such aspect is the extraction of the knowledge and ideas conveyed by patent images. Another important aspect of

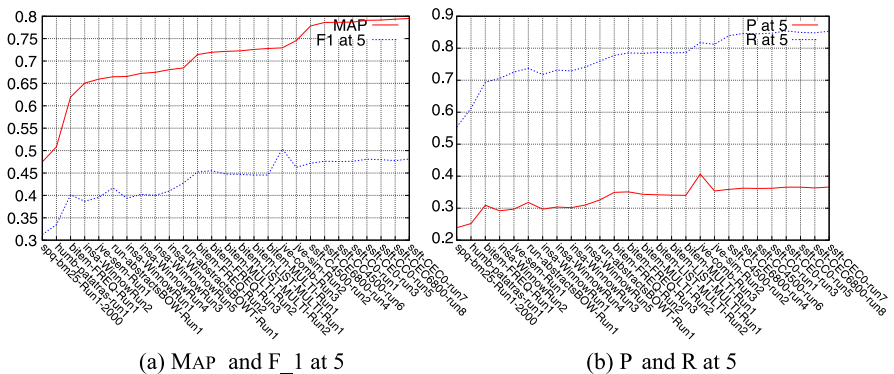
**Fig. 4.4** NDCG and PRES measure values for CLEF-IP 2010, prior art task



patent retrieval, which was not yet addressed by the CLEF-IP campaign, is that information search is session based: the final list of relevant documents is the result of several search queries, possibly built on each other. Both these research directions need sustained support from the IP community.

Nevertheless, along with the TREC-CHEM campaign (described in the following chapter) and the patent oriented campaigns organized in the frame of the NTCIR project [9], the CLEF-IP campaigns actively contribute to raising the IR’s community interest in exploring a body of knowledge that has such a high impact in the economic world.

**Acknowledgements** We thank Matrixware Information Systems GmbH for making available the patent corpus for this track, and for co-organizing the first evaluation campaign. We also thank Judy Hickey and Henk Tomas for sharing their know-how on prior art searches and patent life-cycles with us.



**Fig. 4.5** Evaluation results for the classification task in 2010

## Appendix

**Table 4.2** Indexing the data. Below, x indicates a field is used, – not used, x! indicates special treatment and ? indicates a lack of information on field usage

Group ID	IPC	title	clms	abs	desc	System	Other notes
CLEF-IP 2009							
cwi	x	x	?	?	?	<ul style="list-style-type: none"> <li>• MonetDB/XQuery</li> <li>• PF/Tijah module</li> </ul>	<ul style="list-style-type: none"> <li>• domain specific index containing relations (e.g. inventor and patents, IPC and docs)</li> <li>• domain unaware index</li> <li>• stored in a SQL database</li> <li>• split on 4 different databases</li> </ul>
clefip-deu	x	x	x	x	x	Indri	<ul style="list-style-type: none"> <li>• merged patent docs into one doc, using last versions of fields in docs.</li> <li>• English only</li> <li>• where fields were missing, content in other fields was used to fill them up (abs into desc, or claims; title into abs, desc, claims, etc.)</li> <li>• list of stop words extracted by term freq in fields across all docs</li> </ul>
heuge	x	x	x	x	x	Terrier	IPC once limited to 4 char, once complete
Hildesheim	–	x	x	–	–	Lucene	<ul style="list-style-type: none"> <li>• one English and one German index</li> <li>• German Analyser on German content</li> </ul>
humb	x	x	x	x	x	<ul style="list-style-type: none"> <li>• PATATRAS with MySQL to store metadata</li> <li>• own HMM-based implementation for English content</li> <li>• Tree Tagger for French and German</li> </ul>	<ul style="list-style-type: none"> <li>• further metadata used included patent applicants' name and address, citations</li> <li>• indices built at lemma level; for English an additional phrase index; concept tagging</li> <li>• cross-lingual concept index (multilingual terminological database)</li> </ul>
NLEL	–	x	–	x	x	JIRS (for Passage Retrieval) developed at Univ. Poli. de Valencia	<ul style="list-style-type: none"> <li>• used only one doc from a set of patent documents (the 'last' version/ktind)</li> <li>• 1 per year and language</li> </ul>
clefip-run	–	–	x	–	–	Indri/Lemur	used a stop word list for English

Table 4.2 (Continued)

Group ID	IPC	title	clms	abs	desc	System	Other notes
TUD	x	x	x	x	x	x	<ul style="list-style-type: none"> <li>• preprocessing: UIMA (Unstruc. Infor. Management Archit.) in DKPro Infor Retr framework;</li> <li>• indexing: Lucene</li> </ul>
UAIC	-	x	x	x!	x!	adapted Lucene indexer	<ul style="list-style-type: none"> <li>• desc is used/indexed only if abs is not available</li> <li>• 4 parallel indices (peer-to-peer)</li> <li>• English only</li> </ul>
clefip-ug	x	x	x	x	x	Indri/Lemur	Did not index the separate fields, but the doc as a whole, with a minimal list of stopwords
clefip-unique	x	x	x	x	-	?	<ul style="list-style-type: none"> <li>• English only</li> <li>• used applicant and inventor fields, too</li> </ul>
UniNE	x	x	x	x	x	?	<ul style="list-style-type: none"> <li>• used stemmers and stopwords lists for each language;</li> <li>• one mixed-language index</li> </ul>
uscom	x	x	x	x	x	see clefip-ug	<ul style="list-style-type: none"> <li>• used one multilingual index of clefip-ug;</li> <li>• indexed all patent docs</li> </ul>
UTASICS	x	x	x	x	x	Indri/Lemur	<ul style="list-style-type: none"> <li>• one index per language;</li> <li>• one IPC index truncated at 4 chars;</li> <li>• indexed a virtual patent (with fields taken from the latest patent doc possible)</li> </ul>
CLEF-IP 2010							
bitem	x	x	x	x	-	Terrier	<ul style="list-style-type: none"> <li>• also indexed applicant and inventor fields</li> <li>• one English index</li> <li>• Porter stemming, removed stopwords</li> <li>• Google Translate for document fields not in English</li> </ul>

Table 4.2 (Continued)

Group ID	IPC	title	clms	abs	desc	System	Other notes
deu	x	x	x	x	x	Indri	<ul style="list-style-type: none"> <li>English index only</li> <li>stemming, number and stopwords removal</li> </ul>
hild	x	x	-	x	-	Apache/Lucene	<ul style="list-style-type: none"> <li>English index only</li> <li>Patent UCIDs</li> <li>Porter stemmer, removed stopwords</li> </ul>
humb	x	x	x	x	x	PATATRAS <sup>a</sup>	<ul style="list-style-type: none"> <li>all fields in the patent document were used</li> <li>one index per language, and one concept based index (GRISP)</li> <li>used MySQL database, data cleaning</li> <li>citation mining</li> </ul>
run	x	x	x	x	x	Indri/Lemur	<ul style="list-style-type: none"> <li>one index per document field (title, claims, etc)</li> <li>one index for the full text</li> </ul>
spq	x	?	?	?	?	own indexing system, based on MonetDB/XQuery	participated in 2009 with the same system; participant id: cwi
ui	x	x	x	-	x	Indri/Lemur	<ul style="list-style-type: none"> <li>indexed many other fields of the patent documents</li> <li>no stemmer</li> </ul>
uned	-	x	x	x	x	BM25 based	<ul style="list-style-type: none"> <li>one index per language</li> <li>joined various patent documents of a patent into one document</li> </ul>
uaic							Same system as in 2009, see above

<sup>a</sup>Used Lemur for indexing, Okapi BM25 for lemma indices and concept index creation



**Table 4.3** Query generation, retrieval systems, and ranking. Below, x indicates a field is used, - not used, x<sup>1</sup> indicates special treatment and ? indicates a lack of information on field usage

Group ID	title	clms	abs	desc	IPC	Query generation method	Retrieval system	Ranking results	MT	Other notes
CLEF-IP 2009										
cwi	?	?	?	?	?	<ul style="list-style-type: none"> <li>visual interface, drag'n'drop to create complex search strategies</li> <li>translated into PRA (Probabilistic Relational Algebra), then SQL queries</li> <li>term selection: tf-idf</li> </ul>	<ul style="list-style-type: none"> <li>MonetDB</li> <li>SQL queries</li> </ul>	<ul style="list-style-type: none"> <li>bm25, boolean</li> <li>category (IPC based)</li> </ul>	-	HySpirit software component for PRA is developed by Apriorie
clefip-dcu	x	x <sup>1</sup>	x	x <sup>1</sup>	x <sup>1</sup>	<ul style="list-style-type: none"> <li>topic docs processed as those for indexing</li> <li>abs and desc used only for English topics</li> <li>bi-gram words</li> </ul>	Indri?	Indri?	-	IPC classes used to filter out results
heuge	x	x	x	x	x	Terrier 2.2.1	Terrier 2.2.1	<ul style="list-style-type: none"> <li>Terrier</li> <li>bm25, tf-idf, bb2, etc.</li> </ul>	x <sup>a</sup>	Citations and IPCs used to post-process results
Hildesheim	x	x	-	-	-	Simple term queries	Lucene	Lucene based	-	Later experiments used IPCs and Snowball stemmer.
humb	x	x	x	x	x	<ul style="list-style-type: none"> <li>Topic docs processed as docs for indexing</li> </ul>	<ul style="list-style-type: none"> <li>PATATRAS</li> <li>Lemur toolkit</li> <li>Unigram, KL, Okapi</li> </ul>	<ul style="list-style-type: none"> <li>result lists merged using SVM regression models and linear comb of normalized ranking scores</li> <li>post-ranking using SVM regression model</li> </ul>	x <sup>b</sup>	<ul style="list-style-type: none"> <li>reduced topic search space by using working sets per topic.</li> <li>citations, applicant address, name, ECLA codes</li> </ul>

<sup>a</sup>Google Translate for the language tasks where fields were missing

<sup>b</sup>Multilingual terminology database

Table 4.3 (Continued)

Group ID	title	clms	abs	desc	IPC	Query generation method	Retrieval system	Ranking results	MT	Other notes
NLEL	x	-	x	-	-	Summarization by random walks, from the topic original language (only claims were for sure in all lang)	Adapted JIRS for Passage Retrieval		x <sup>c</sup>	One query per language
clefip-run	-	x	-	-	-	Complete claims	Lemur	tf-idf	-	-
TUD	x	x	-	-	x!	<ul style="list-style-type: none"> <li>• IPC as a separate query</li> <li>• one query per lang</li> <li>• min 800 words (from title, clms and abs or desc when claims field was shorter)</li> </ul>	Lucene based	Lucene based (tf idf?)	-	-
UAIC	x	x	x!	x!	-	Lucene based	Lucene	Lucene based, with boost factors		desc is used/indexed only if abs is not available
clefip-ug	x	x	x	x	x	<ul style="list-style-type: none"> <li>• df, tf, query length thresh-olds;</li> <li>• 1: computes distribution of terms to related patents</li> <li>• 2: based on this distribution, does query extraction from the topics</li> </ul>	Indri/Lemur	bm25	-	-
clefip-unige	x	x	x	x	x	whole patent?	?	tf-idf, Okapi, Fast	-	<ul style="list-style-type: none"> <li>• filtering of results by IPC class</li> <li>• filtering on the length of docs</li> </ul>

<sup>c</sup>Google Translate

Table 4.3 (Continued)

Group ID	title	clms	abs	desc	IPC	Query generation method	Retrieval system	Ranking results	MT	Other notes
UniNE	x	x	x	x	x	tf-idf; max 100 terms	?	tf-idf with cosine normalization; BM25, DFR (divergence from randomness) LM (stat language model)	–	no workshop notes
uscom	x	x	x	x	x	<ul style="list-style-type: none"> <li>• idf</li> <li>• extracted query terms per language, then compiled them</li> <li>• no. of query terms: a) fixed; b) percentage of topic length</li> </ul>	bm25 (Lemur)	<ul style="list-style-type: none"> <li>• given by bm25</li> <li>• the patent doc highest in result list was kept</li> </ul>	–	
UTASICS	x	x	x	x	x	<ul style="list-style-type: none"> <li>• ratf term selection before Google Translate</li> <li>• tf-idf</li> <li>• max 50 terms</li> </ul>	Indri (Lemur)	<ul style="list-style-type: none"> <li>• rank given by Indri on the 4<sup>x</sup> queries and</li> <li>• Mean Average Distance to combine query results (involves relative weight computation)</li> </ul>	<ul style="list-style-type: none"> <li>• 3 queries (one per lang) + one IPC query</li> <li>• various combinations of fields (abs + desc, abs + claims, etc)</li> <li>• additional manual queries with up to 10 keywords (IP experts involved)</li> </ul>	
CLEF-IP 2010										
bitem	x	x	x	–	x	Terrier with Bo1 query expansion	Terrier	Terrier PL2 weighting	x <sup>e</sup>	exploited citations and IPC codes in the post-processing phase

<sup>d</sup>Google Translate to get query terms where fields missed content in the query language

<sup>e</sup>Google Translate into English

Table 4.3 (Continued)

Group ID	title	clms	abs	desc	IPC	Query generation method	Retrieval system	Ranking results	MT	Other notes
deu	x	x	x	x	x	<ul style="list-style-type: none"> <li>uni-grams extracted from desc</li> <li>bi-grams extracted from the complete text, with frequency &gt; 3</li> </ul>	Indri	Language models and Inferred networks	x	exploited citations
hild	x	x!	x	x	x	Apache/Lucene	BM24, Okapi		-	<ul style="list-style-type: none"> <li>used the main claim only</li> <li>Porter stemmer for stopword removal from topic files</li> </ul>
humb	x	x	x	x	x	used all textual content	PATATRAS (see entry for 2009 above)	BM24 & Indri, SVM reranking	x <sup>f</sup>	<ul style="list-style-type: none"> <li>citation analysis - key term extraction from scientific articles (GROBID)</li> </ul>
run	x	x	x	x	x	Lemur	tf-idf, Logistic models	regression	-	
spq	x	x	x	x	x	<ul style="list-style-type: none"> <li>graphical strategy building</li> <li>Snowball stemmer</li> <li>tf-idf selecting first 26 terms</li> </ul>	Own	BM25		participated in 2009 as cwi
ui	x	x	x	x	x	tf-idf selecting top 10 terms	Lemur/Indri	Lemur/Indri	-	
uned	x	x	x	x	x	term selection based on language models	BM25	KLD (Kullback-Leibler divergence)	Di-	-
uaic						boost values				Same system as in 2009, see entry above

<sup>f</sup>Multilanguage concept database (GRISP)

**Table 4.4** Systems, methods and document fields used in the Classification task

Group ID	Training/Classification fields	Preprocessing	Classification systems	Ranking	MT	Other notes
bitem	title, abstract, claims, description, applicant, citations	Porter stemmer, Stopword removal (English only)	re-kNN	BM25, BM25/DFR and PL2	x	Query translation as in the PAC task. Retrieval step, results sent to the classifier.
humb	all, except legal information and ICO classification codes	same as in PAC, no concept tagging	re-kNN, co-classification	see PAC entry	x	retrieval step followed by a classification step.
insa	abstract, title, description, names, address	extracting bag of words and linguistic triples with AGFL <sup>a</sup>	LCS2 <sup>b</sup> , balanced Winnow	system based	–	different runs were obtain based on different set of training fields used.
jve	title, description, claims, abstract	POS and key–phrase tagging WordNet	<ul style="list-style-type: none"> <li>• SVM</li> <li>• Lemur</li> <li>• combined</li> </ul>	system based	–	<ul style="list-style-type: none"> <li>• SVM was trained on each language</li> <li>• first run obtained with SVM</li> <li>• second run obtained with Lemur (indexing and retrieval)</li> <li>• third run combined the first two.</li> </ul>
run	abstracts	<ul style="list-style-type: none"> <li>• remove punctuation, numbers</li> <li>• make all text lowercase</li> <li>• simple tokenization</li> <li>• dependency triples obtained with AEGIR parser</li> </ul>	LCS Winnow with Lucene analyzer	system based	–	only English content was used.
spq	see PAC entries				–	the same system was used as in the PAC task, the IPC codes of the retrieved documents were given as results.
ssft	inventor, applicant, title, abstract, claims, description	collocation extraction during indexing	myClass - Winnow-like in-house implementation	system based	–	<ul style="list-style-type: none"> <li>• when large, the description field was limited to 4K</li> <li>• additional patent data from other sources was used for some of the runs</li> </ul>

<sup>a</sup><http://www.agfl.cs.ru.nl><sup>b</sup><http://www.phasar.cs.ru.nl/LCS/>

## References

1. Conference on Multilingual and Multimodal Information Access Evaluation (2010). <http://clef2010.org/>
2. Cross Language Evaluation Forum. <http://www.clef-campaign.org>
3. European Patent Convention (EPC). <http://www.epo.org/patents/law/legal-texts>. URL <http://www.epo.org/patents/law/legal-texts/epc.html>
4. Fujii A, Iwayama M, Kando N (2007) Overview of the patent retrieval task at the NTCIR-6 workshop. In: Kando N, Evans DK (eds) Proceedings of the sixth NTCIR workshop meeting on evaluation of information access technologies: information retrieval, question answering, and cross-lingual information access. National Institute of Informatics, Tokyo, pp 359–365
5. Graf E, Azzopardi L (2008) A methodology for building a patent test collection for prior art search. In: Proceedings of the second international workshop on evaluating information access (EVIA)
6. Guidelines for Examination in the European Patent Office (2009). <http://www.epo.org/patents/law/legal-texts/guidelines.html>.
7. Järvelin K, Kekäläinen J (2002) Cumulated gain-based evaluation of IR techniques. *ACM Trans Inf Syst* 20(4):422–446
8. Magdy W, Jones GJF (2010) PRES: A score metric for evaluating recall-oriented information retrieval applications. In: SIGIR
9. NTCIR Project (2010) Evaluation of information access technologies research infrastructure for comparative evaluation of information retrieval and access technologies. <http://research.nii.ac.jp/ntcir/index-en.html>
10. Peters C, Di Nunzio G, Kurimo M, Mostefa D, Penas A, Roda G (eds) (2010) Multilingual information access evaluation I. Text retrieval experiments. Lecture notes in computer science, vol 6241. Springer, Berlin
11. Piroi F, Tait J (2010) CLEF-IP 2010: Retrieval experiments in the intellectual property domain. Tech Rep IRF-TR-2010-0005, Information Retrieval Facility, Vienna, Austria. URL <http://www.ir-facility.org/research/publications-reports/technical-reports/files/irf-tr-2010-00005.pdf>
12. Piroi F, Roda G, Zenz V (2009) CLEF-IP 2009 evaluation summary. Tech Rep IRF-TR-2009-00001, Information Retrieval Facility, Vienna, Austria. URL [http://www.ir-facility.org/research/technical-reports/files/irf\\_tr\\_2009\\_00001.pdf](http://www.ir-facility.org/research/technical-reports/files/irf_tr_2009_00001.pdf)
13. Roda G, Tait J, Piroi F, Zenz V (2010) CLEF-IP 2009: Retrieval experiments in the intellectual property domain. In: Peters C, Di Nunzio G, Kurimo M, Mostefa D, Penas A, Roda G (eds) Multilingual information access evaluation I. Text retrieval experiments. Lecture notes in computer science, vol 6241. Springer, Berlin, pp 385–409. doi:10.1007/978-3-642-15754-7\_47
14. Suzan Verberne Eva D’hondt, NOCHK
15. Text Retrieval Conference. <http://trec.nist.gov>
16. The MAtrixware REsearch Collection (2010). <http://ir-facility.net/prototypes/marec/description/overview/>

# Chapter 5

## Evaluation of Chemical Information Retrieval Tools

Mihai Lupu, Jimmy Huang, and Jianhan Zhu

**Abstract** It has been noted before in this book that patent retrieval is different from, and more complicated than “standard” information retrieval. Evaluation of patent retrieval engines has also been shown to require specific attention. In this chapter, we continue making this point, but emphasize the efforts undertaken in a specific domain, namely chemistry. We approached this issue from two different perspectives. First, there is the issue of scalability. Largely similar to the CLEF-IP efforts, it targets the problem of having to handle a large number of documents and, potentially, a large number of queries. Second, there are the issues generated by the specific characteristics of chemistry documents. We describe here how we manually created a set of topics to reflect the kind of requests for information that a patent searcher, or a general researcher, might have. The results of the first year’s track are presented as well, together with directions and desiderata for the next years.

### 5.1 Introduction

The interest in domain-specific information retrieval, and therefore in IR evaluation, is certainly not new. Particularly in the bio-domain there has been considerable interest and support from governmental, supra-governmental and private actors. The TREC Genomics track [4] started in 2002 and ran for 4 years. It aimed at tackling the problem of the rapidly increasing *bibliome* or literature of biology, as a result of new experimental methods in the biomedicine field, which generate far larger amounts of data than ever before (e.g. microarrays or sequencing technology). A year later, in

---

M. Lupu (✉)  
Information Retrieval Facility, Vienna, Austria  
e-mail: [m.lupu@ir-facility.org](mailto:m.lupu@ir-facility.org)

J. Huang  
York University, Toronto, Canada  
e-mail: [jhuang@yorku.ca](mailto:jhuang@yorku.ca)

J. Zhu  
True Knowledge Ltd., Cambridge, UK  
e-mail: [jianhanzhu@gmail.com](mailto:jianhanzhu@gmail.com)

2003, the BioCreative [5] challenge started the work on *a community-wide effort for evaluating text mining and information extraction systems applied to the biological domain*. Both of these campaigns have played an essential role in demonstrating the interest and the need to have dedicated IR tools, and therefore evaluations, for specific areas of research.

The TREC Chemical IR Evaluation campaign (TREC-CHEM) follows partially on the footsteps of these two previous campaigns. However, it takes a more holistic view and attempts to promote a merger of very domain-specific methods of chemical retrieval (i.e. structure search) and general text mining and information retrieval methods.

## Objectives

The objectives of TREC-CHEM are:

1. to provide a reference corpus for the special domain of chemistry, covering a wide scientific area both in terms of sub-domains and publication venues
2. to promote comprehensive and scalable approaches to chemistry information retrieval, which combine structure search and text-based IR methods
3. to introduce academic and industry researchers to a new category of scientific articles: patent documents.

In the next sections we will describe how we approached these objectives. We start with a large section describing the test collection, including the documents, the topics and the procedures of obtaining the relevance judgments (Sect. 5.2). Section 5.3 provides a summary of the participants' methods and their results in the 2009 campaign. We conclude with observations and future directions in Sect. 5.4.

## 5.2 Test Collection

The nature itself of experimental evaluation campaigns give paramount importance to the test collection. From the first book [7], up to the latest articles [13], the issue has been a problematic one. In general, we have followed a traditional approach, based on the existing experimentation standards described in the above references, but also in the preceding chapters. With the aim of the campaign being the evaluation of holistic IR methods in the chemical domain, we did not, at least in the first two years, target specifically any form of structure search or chemical reaction search. Therefore, the test collection did not contain annotated documents, information extraction topics, or relevant objects other than full documents. Each of them is described in what follows.



## 5.2.1 Documents

The first objective of the campaign is to provide a large corpus of chemistry-related documents. To do this, we benefited from the support of the Royal Society of Chemistry (RSC), who provided articles from over 30 journals spanning a period from 1997 to 2006. We expanded this collection of scientific articles in the 2010 collection by adding other Open Access publications, many from PubMed Central [12], but also from other publishers, such as IUCR Journals,<sup>1</sup> Hindawi Publishers,<sup>2</sup> Oxford University Press<sup>3</sup> and Molecular Diversity Preservation International.<sup>4</sup>

In addition to this, both the 2009 and the 2010 corpora contain over 1 million full text patent documents from the EPO and USPTO in 2009, plus the WIPO in 2010.

### 5.2.1.1 Patent Documents

Both in 2009 and 2010, the collection of patent documents used consisted of XML files created by Matrixware Information Services GmbH. Regardless of the original source of the patent data, all XML files followed the same DTD,<sup>5</sup> which made things particularly convenient for the participants. In terms of chemistry-specific issues, these DTDs did contain `<chemistry>` tags, but these were not consistently used throughout the collection and therefore could not be relied upon. Furthermore, a lot of chemistry-specific information was present in attachments to the documents, which were not available in the 2009 corpus, but were made available in the 2010 one. Attachments take many forms, but the most common ones were TIF images as well as MOL and CDX structure information files.

In terms of covering the different areas of chemistry, Fig. 5.1 shows the number of patent documents which have at least one of their IPC classes in the class listed on the horizontal axis. As expected, the most pro-eminent classes are A61 (mainly A61K—*Preparations for medical, dental, or toilet purposes*), C07—*Organic chemistry*, C08—*Organic macromolecular compounds*, C12—*Biochemistry, beer, spirits, wine, vinegar, microbiology, enzymology, mutation or genetic engineering*.

### 5.2.1.2 Scientific Articles

In 2009, the scientific journals provided by the RSC, 31 of them, covered more or less equally the different areas of chemistry. With the introduction, in 2010, of the

---

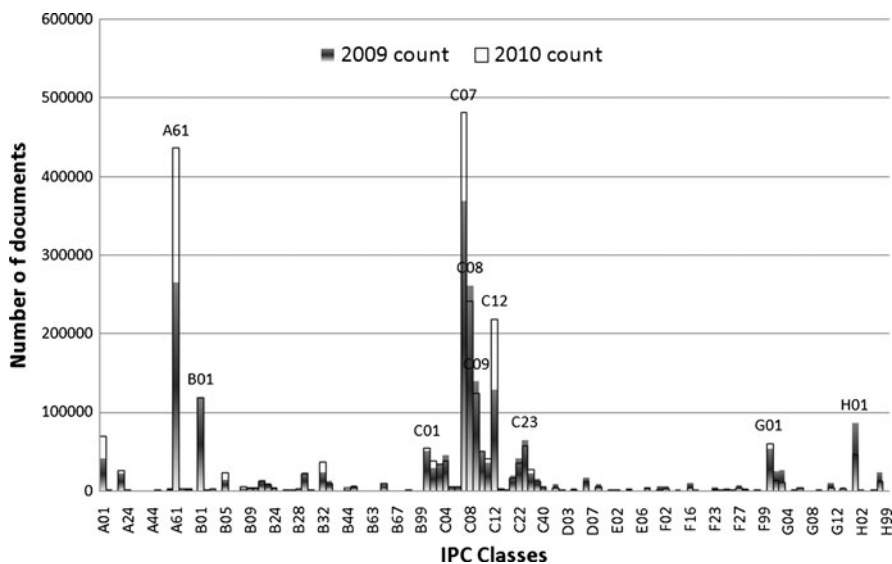
<sup>1</sup><http://journals.iucr.org/>.

<sup>2</sup><http://www.hindawi.com/>.

<sup>3</sup><http://ukcatalogue.oup.com/>.

<sup>4</sup><http://www.mdpi.org/>.

<sup>5</sup>The MAREC DTDs are publicly available together with the MAREC data, conditioned on the signing of a license agreement.



**Fig. 5.1** Distribution of IPC classes across the TREC CHEM 2009 and 2010 collections

large corpus from PubMed Central Open Access journals, there is some bias toward biomedical and life sciences journal literature. We tried to compensate for this by including publications from individual publishers, but PMC is still outweighing the others two-fold (approximately 120 000 articles from PMC versus approx 60 000 from individual publishers, including RSC). However, the focus on the biomedical and life sciences part of chemistry is something that we have observed in patents, as well as in the topics provided by experts. Ultimately, it reflects a general interest of our society in this area of research. Therefore, we considered that the advantages of including this set of articles will compensate for the possible disadvantages.

## 5.2.2 Topics

If the corpus of the test collection is the foundation on top of which the entire campaign is built, the topics are the seeds that give the track its direction. They are the means to achieve the second objective of the campaign: promoting comprehensive and scalable approaches to chemistry information retrieval, combining structure and text-based IR methods.

In designing the topics for the 2009 and, later, for the 2010 campaigns, we considered the following goals and constraints:

Goals	Constraints
<ul style="list-style-type: none"><li>• cover a wide area of chemistry sub-domains in order to discover particular problems in different such sub-domains</li><li>• address potential participants from both text retrieval research groups and chemo-informatics groups</li><li>• consider the scalability of the methods proposed</li><li>• make the evaluation resilient to abnormalities that may be present in the topic set</li></ul>	<ul style="list-style-type: none"><li>• be specific enough to attract interest from specialists, who generally work in one of the sub-domains of chemistry</li><li>• give the text part of the retrieval process sufficient weight to make it interesting for most IR groups, but without making the chemo-informatics groups lose interest</li><li>• generate a Gold Standard for all topics</li></ul>

To reconcile these goals and constraints, we split the campaign into two tasks. We called them, more or less arbitrarily, the “Prior Art” (PA) task and the “Technology Survey” (TS) task. The first is designed to work with a large number of topics (1000) in order to get some statistical significance of the results, while the second one is designed to work with considerably fewer topics (we had 18 in 2009 and 30 in 2010), but to provide a better understanding of what exactly works and what does not work in each system. The next two sections describe them both.

### 5.2.2.1 Patents as Topics

The “Prior Art” task attempts to reproduce the list of citations that accompany a patent application or a granted patent. In this sense, they do not necessarily have to invalidate the claims listed in the patent application. In this case, a topic is simply a request to produce references to patents which may be considered by an expert for further analysis. An example of such a topic follows:

```
<topic>
  <num>PA-3</num>
  <title>4-Tetrazolyl-4-phenylpiperidine derivatives for
    treating pain</title>
  <narr>Find all patents in the collection that would
    potentially invalidate patent application
    EP-1803718-A1 </narr>
  <file>EP-1803718-A1.xml</file>
</topic>
```

Figure 5.2 shows the distribution of the IPC classes across the 1000 PA topics in the 2009 and 2010 collections. As expected, since the topics were extracted from the corpus randomly (subject to constraints), the distribution shown in Fig. 5.2 matches

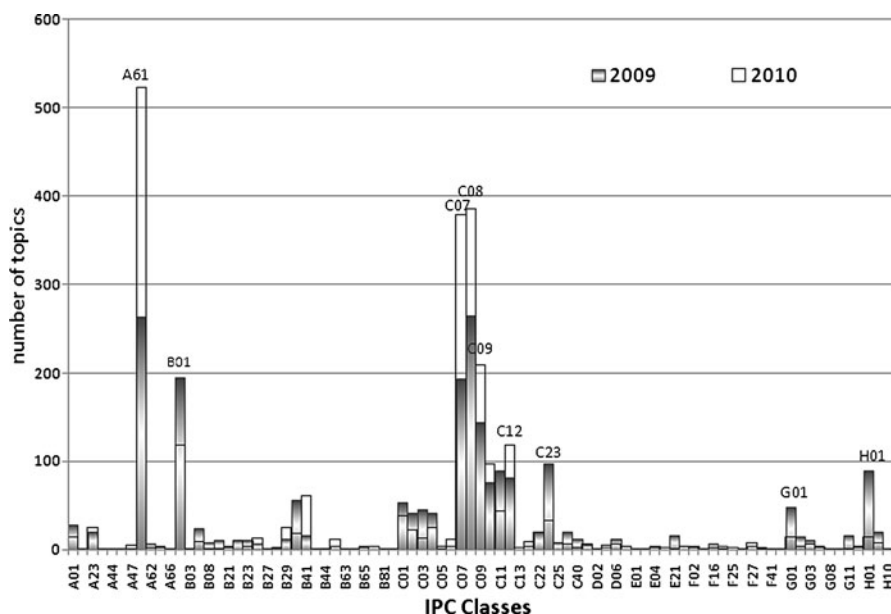


Fig. 5.2 Distribution of IPC classes across the TREC CHEM 2009 and 2010 PA topics

the distribution in Fig. 5.1. We also observe that in 2010 the set of topics is more focused on the more prevalent classes (A61, C07, C08, C09). This, however, is not the result of a decision to favor these classes, but more likely a consequence of the random selection.

We should note that, as the reader may have already observed, the sum of the values of the columns in both Fig. 5.1 and 5.2 is greater than the total number of documents, and, respectively, the total number of topics. This is due to the fact that a patent can be classified in more than one IPC class and it is in fact rather common that a patent in A61K (medical, dental, toiletries) may also have a classification in Section C (Chemistry).

In retrieving prior art candidates for the given patent documents, the participating research groups are free to use any part of the patent and any method to extract the most relevant keywords or to train their system using whatever means necessary. The only restriction imposed on the participating systems is not to use the citations of the topic patent documents directly in their results. In Sect. 5.2.3.1 we shall see exactly why.

### 5.2.2.2 Manual Topics

While the PA, “automated”, topics provide a way to test the IR systems on a large scale and to obtain some statistical significance of the results by testing the systems over a large number of topics, it is easy to fall into the traps of over-averaging results:

we average over the measure values at different cut-offs, over different topics, over different sets of topics or different runs, and in the end, it is easy to lose track of what is actually going on. This is why we introduced the “Technology Survey” (TS) topics: to let the systems be tested and inspected manually, by experts in the field, for topics that are as close as possible to their line of work.

A TS topic is a general request for information, formulated as naturally as possible. Here are two examples:

```
<topic>
<number>TS-2</number>
<kind>pharmaceuticals</kind>
<title>Dipetidyl peptidase-IV inhibitors</title>
<narrative>
We are a new pharmaceutical company that is interested in
entering the area of Dipetidyl peptidase-IV inhibitors.
This is a relatively new therapeutic area for the treatment
of type-2 diabetes but we know that there are compounds
already generated by several pharmaceutical companies
(including a marketed drug from Merck called Januvia) for
this indication. We are interested in discovering the
compounds that have been identified so far for inhibiting
this enzyme and which companies they are associated with.
It would also be useful to determine if there is more than
one chemical class of compounds that is used to inhibit
this enzyme or if several classes have been identified.
</narrative>
<expert>Anthony Trippe</expert>
</topic>
```

```
<topic>
<number>TS-5</number>
<kind>organic, high molecular weight</kind>
<title>
(Pregna-4,17-diene-3,16-dione or Guggulsterone
or RN:95975-55-6)
</title>
<narrative>
Documents on (Pregna-4,17-diene-3,16-dione or Guggulsterone
or RN:95975-55-6)---particularly on preparation
</narrative>
<expert>anonymous2</expert>
</topic>
```

As can be seen, each topic consists of three main parts: *title*, *kind* and *narrative*, as well as two ‘organizational’ tags: *number* and *expert*. The *narrative* describes, in

**Table 5.1** Details possibly present in a 2010 TS topic

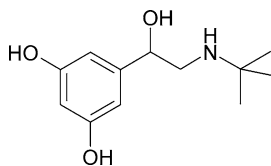
Tag	Description	Example
Chemicals	one or more chemical compounds	Dehydroepiandrosterone
Reactions	a chemical reaction	carbon-carbon coupling
Administration	in the case of pharmaceuticals, how is the compound administered	oral
Condition	a disease or other medical condition	tooth decay
Target	particularly in the case of pharmaceuticals, the component that the chemical affects	D-alanine-D-alanine ligase

a language as natural as possible, a request for information. We advised the creators of the topics to start with “*We are a group of researchers doing...*” to make sure that the reader understands that this is a fairly generic requests for information. The *title* and *kind* are there to summarize the content of the narrative and to provide some context. However, particularly the *kind* tag, they were not really used by the participants.

In 2010 we changed this structure slightly, in an attempt to provide more clues to the searcher as to what exactly is being searched for. Here is an example of a 2010 TS topic:

```
<topic>
<number>TS-25</number>
<title>Methylene phosphonic acid as flame retardant</title>
<narrative>
We are looking for patents on the use of methylene
phosphonic acid as a flame retardant.
</narrative>
<details>
<chemicals>Methylene phosphonic acid</chemicals>
</details>
<relevance>
A document will be considered RELEVANT if it refers
to the use of ANY Phosphonates or phosphonic acids
specifically as a flame retardant
A document will be considered HIGHLY RELEVANT when it is
RELEVANT and it is specifically methylene phosphonic acid
used.
</relevance>
</topic>
```

We discarded the *kind* tag and introduced instead a *details* tag, which may contain a number of different sub-tags: *chemicals*, *reactions*, *administration*, *condition*, *target*. Table 5.1 describes these detail tags.



**Fig. 5.3** Example of the chemical compound referenced in the 2010 TS-46 topic. This compound is provided to the participants both as an image, and as a structure file (MOL file)

In addition to these topics, in 2010 we also introduced two topics to nudge the participants toward a more structure-based search. The text of these topics is quite simple at this time, asking the users to provide documents that refer to a chemical whose formula is provided both as an image, as well as a MOL file. Figure 5.3 shows the chemical compound referenced by topic TS-46, listed below:

```
<topic>
<number>TS-46</number>
<title>Structure search 1</title>
<narrative>
We are looking for patents or scientific articles on
methods of preparation of the compound described in
TS-46.mol and TS-46.png
</narrative>
<details>
</details>
<relevance>
A document will be considered RELEVANT if it describes
methods of preparation of the given compound. It shall not
be considered relevant if it describes derivatives of this
compound.

There are no HIGHLY RELEVANT documents.
</relevance>
</topic>
```

There is arguably a lot of space for improving the definition of such topics, but the aim until now has been to introduce this kind of problem to the participants and allow them to familiarize themselves with this problem. The search methods for such searches are very different from the kind of text searches that most of our participants are used to, and we encourage them, starting with 2010, to combine text-based search and structure-based search. At the same time, such topics aim to bring the topics closer to real life practice and to attract the research groups who are proficient in structure search only. Ultimately, a collaboration of these two kinds of groups is highly desirable.

### 5.2.3 Relevance Judgments

The last, but equally important part of an evaluation test collection, are the relevance judgments: the function that, given any pair consisting of a request for information and a document, would return a relevance value. This part of a test collection is, intuitively, extremely important. At the same time, it has been observed before [15] that, while inconsistencies are always present, a sufficiently large number of topics would generally be able to consistently identify the best performing systems and where inversions in the rankings of the systems occur, it is only for pairs of systems that, across a series of experiments, return similar results.

For chemistry, the difficulty of creating relevance judgments lies mainly in the fact that assessors must be qualified professionals. Apart from this, one could argue both ways with regards to the difficulty or easiness of creating relevance assessments: on one hand some topic could be hard to evaluate because generic compounds may introduce ambiguities. On the other hand, the exact science nature of the field would arguably induce more precise relevance judgments than it is the case in 'general' IR.

All the other problems of creating relevance judgments presented in previous chapters are of course still valid in this domain: impossibly large sets of results, and incompleteness of relevance judgments.

In what follows we will present the relevance judgments for the two tasks of the TREC-CHEM track.

#### 5.2.3.1 Citation Based Relevance Judgments

The Prior Art task, which concerns itself only with patent data, has relevance judgments based on citations in the topic patents, created in the same way as described in the CLEF-IP Chapter above, and therefore we will not go again into details here. Instead, Fig. 5.4 shows some statistics regarding the number of citations per topic, which are considerably different from those of the CLEF-IP track, mainly due to the presence of US patents in our test collection.

#### 5.2.3.2 Manual Relevance Judgments

For the Technology Survey (TS) task, the relevance judgments were much more difficult to create. As mentioned in Sect. 5.2.2.2, the aim of this task is to simulate as accurately as possible a real life request for information. If that is to be achieved, bypassing manual judgments is impossible. We therefore set forth to create these relevance judgments, with the help of the experts that created the topics and of chemistry graduate students.

In 2009, the procedure to obtain relevance judgments involved a standard pooling technique followed by a stratified sampling to select a manageable set of documents



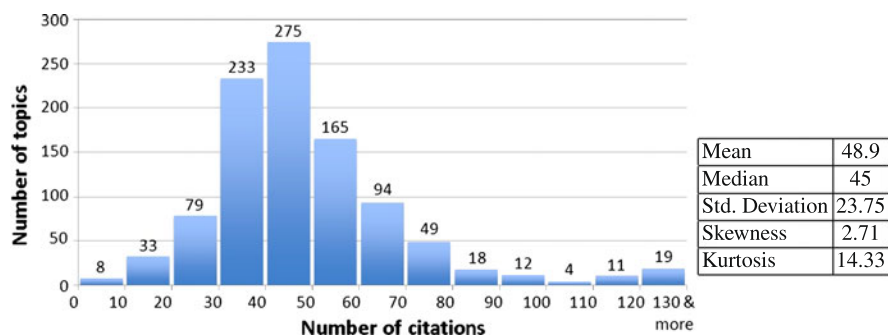


Fig. 5.4 Statistics for the 2009 PA qrels sets

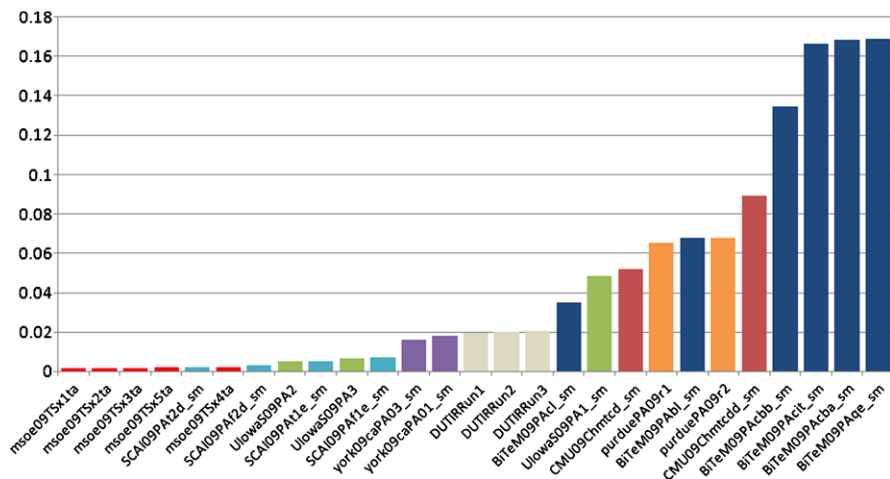
to evaluate manually (on average, 300 documents per topic). The sampling technique also determined the measures we could compute, namely MAP and NDCG, following the method introduced by Yilmaz et al. [16]. In total, for the 18 topics available in 2009, a total of 5518 documents have been manually inspected, out of which, in the end, 941 (17.06%) were found to be highly relevant, 844 (16.30%) relevant, 37 (0.67%) were undecided, 1 not judged and the rest, 3694 (66.96%) not relevant to their respective topic.

These 5518 documents were first viewed in parallel by two students. Then, their results were merged and given to the expert that creating the topic for a final review. In the process, the inconsistencies observed between the two students fell into expected values: 1083 (19.63%) were conflicting decisions (i.e. one student indicated a document as *‘relevant’* or *‘highly relevant’* while the other as *‘non relevant’*), meaning that the rest, over 80% agreed at least in principle (i.e. *‘relevant’* vs. *‘highly relevant’*) or were undecided (i.e. at least one student marking a document as *‘unsure’* or leaving it as *‘unjudged’*).

However, despite the hundreds of hours of work put in by the students, some experts ultimately decided not to use their suggestions and completely re-did the evaluations. We found this to be an unacceptable waste of resources, and for the 2010 campaign we have modified the evaluation interface to allow the student evaluator to interact anonymously with the expert via an exchange of messages. We shall monitor and report on the efficiency of this new method.

### 5.3 Participants and Results

Finally, in this section we will look at the results of the first year, the only ones available at the time of writing of this chapter. We begin with a description of these results, and continue, in Sect. 5.3.2 with a list of very brief descriptions of the participants’ systems, to give us an overview of the approaches used in the first year.

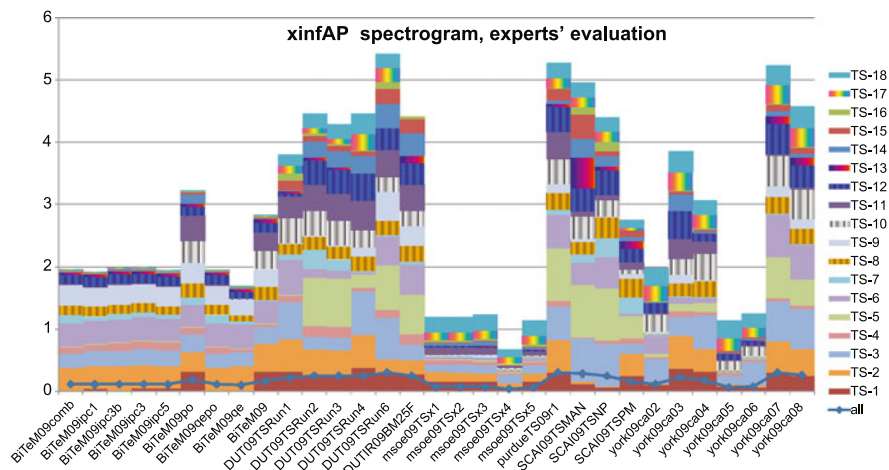


**Fig. 5.5** (Color online) MAP for PA topics. Each *bar* represents the Mean Average Precision of one run (one set of results submitted by a participant). Each *color* represents a different research group

### 5.3.1 Summary of Results

There are many ways to cut the data that were available to us at the end of the 2009 track. 1000 PA topics could be analyzed according to many metrics, grouped by IPC class, number of citations, length of the document, source of the patent, kind of patent document, etc. We shall not go into many details here, but instead we refer the reader to the report of the 2009 track [8], as well as to other papers describing these results [9, 10]. We computed six metrics: Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (NDCG), Binary Preference (bpref), precision at 30 (P@30), recall at 100 (R@100), and Mean Reciprocal Rank (MRR). The best performing system was consistent across all measures, while the worst performing system varied. In 2009, the best performing system managed to retrieve, on average, eight relevant documents in the top 30 results, while ranking 34% of all the relevant documents in the top 100. A summary of the results, for one of the metrics used (MAP) is shown in Fig. 5.5 for all the systems that submitted the results for at least 100 of the 1000 PA topics.

Interestingly, the system that performed best in the PA task did not also perform best in the TS task. That is most likely because in the PA task one could take advantage of features of the patent domain (existing classification, citation networks, priority dates, etc.) which were not available in the collection consisting of scientific articles. Therefore, the systems that took a more generic approach tended to perform better in the TS task. This is visible in Figure 5.6, where the results for one of the two metrics used in the TS task is shown. To note that the metrics used in this task are *inferred* versions of the MAP and NDCG methods, since relevance judgments are not complete.



**Fig. 5.6** “Spectrogram” of TS results according to the extended inferred AP. Each *bar* is a stack of results for each run: the results of each topic are cumulated to form the bar. This is more informative than an average, as it allows the reader to observe differences within a run between different topics

### 5.3.2 Participants

In this section we briefly describe the systems used by the participants to the 2009 track. This helps us get an overview of what works and what does not work, based on their results. References are given to each of their reports.

**Univ. of Applied Sciences Geneva—BiTeM** This was the best performing system for the PA task in the 2009 campaign. The method that proved most successful, and which eventually gave this team the edge, was the use of a learning algorithm that was trained on all the citations available in the patent collection [2]. Despite the fact that citations are also used for the final evaluation, we considered the method valid because it used publicly available data from historic patent collections to train a system to learn associations of terms and documents. The team also used domain-specific entity recognition and IPC filtering, but observed only marginal improvements (albeit consistent improvements). The fact that their biggest improvement came from exploiting citation analysis, led to relatively poor results in the TS task, where what might have been learned from the patent corpus did not apply to the scientific articles. Based on existing observations that the language of the two kinds of documents is different, we can safely assume that the model learned only for patents lead to poor results when applied to the scientific articles.

**Carnegie Mellon University** The CMU team built their retrieval engine on top of their Lemur/Indri toolkit [18]. They only participated in the PA task, where they observed that, using the structured retrieval support provided by Lemur/Indri, using terms from the entire body of the patent document improved results when compared

to using terms only from the title and claims. However, the group did not use any chemistry-specific tools.

**Dalian University of Technology** The IR Group at the Dalian University of Technology also used the Lemur/Indri toolkit, but participated in both the PA and TS tasks [6]. For the PA task they obtained significantly lower results than the CMU group. However, for the TS task they obtained some of the best results. In both cases, they also did not use any chemistry-specific methods, but rather generic methods, including language modeling and pseudo-relevance feedback.

**Iowa University** The Iowa team [11] used another open source IR engine, namely Lucene. They submitted results only for the PA task, for which they used a combination of different fields in the patent (title, abstract, description or claims) and IPC filtering. One of their submissions performed comparably to the CMU system for some of the metrics, but in another set of results they assumed patent numbers to be reflective of a temporal sequence, which is in general not the case, and therefore obtained very low results. Again, no chemistry-specific information extraction method was used.

**Milwaukee School of Engineering** Jay Urbain and Ophir Frider [14] proposed a very interesting distributed system that attempts to index all chemical information found in the patents and scientific articles, and submitted results for both the PA and TS tasks. They used a dimensional index of PubChem terminology for synonym identification, which ultimately achieved good result. However, their official runs contained some errors and were consequently scored very low. This is, however, one of the systems to follow in the upcoming years.

**Purdue University** The system used by the group at Purdue University [1] is particularly interesting because, like the CMU and the Dalian group mentioned above, they have used the Indri search engine, but unlike them, they have also used a chemistry-specific method (i.e. synonyms from PubChem). For the PA task this system performed comparably to the CMU system, and much better than the Dalian system. For the TS task, they performed very slightly worse than Dalian according to one measure (MAP) and better than them according to another measure (NDCG). Given that NDCG favors relevant documents being retrieved early in the ranked list provided by the search system, we would be justified in saying that using chemical information helps achieve better results. However, as also demonstrated by the Geneva group, this improvement, at least for the topics defined in the 2009 track is marginal.

**Fraunhofer SCAI** The SCAI group had some of the best results for the TS task and some of the worst in the PA task [3]. Their method used entity recognition algorithms and automated generation of noun phrases, but also, for the TS task, they submitted results of manually created queries.<sup>6</sup> In some sense, this is the opposing

---

<sup>6</sup>To note that we refer to as a ‘topic’ what we give the participants, and as a ‘query’ what they actually put into their system to obtain results.

system to the one used by the Geneva group: a non-patent specific system performed better on the TS than on the PA task, as opposed to one that takes heavily into consideration the patent corpus and therefore performs better in the PA than in the TS task.

**York University** The York University group [17], who were also co-organizers of the track, participated with a system that tackled two important issues: chemical synonyms and abbreviations. For the PA task, their emphasis was on the extraction of queries from the patent topics (i.e. automated selection of important keywords). When compared to the other results submitted for this task, it would suggest that such a pre-processing step may not in fact be necessary. However, for the TS task, it was very interesting to observe that their use of PubChem data produced a significant improvement in the scores.

## 5.4 Observations and Future Directions

Comparing the different systems that provided results in the 2009 track, and the results they obtained, we can conclude that using chemistry-specific methods does help, but only marginally when the generic method is already very good. We observed such a marginal increase by looking at the results of the University of Applied Sciences, Geneva group, and also by comparing the results of the Purdue group to that of the Dalian group. However, when the generic method does not perform very well, the improvement given by the chemistry-specific methods is significant (we can observe this in the TS runs of the York University group).

The two tasks, though similar in their focus on chemistry, are more different than the organizer had initially thought. Generic text-analysis methods perform well in the PA task and, arguably, using highly specialized methods in this case is not expected to provide great improvements due to, first, the wide area of chemistry covered by the topic set and the patent collection and, second, the verbosity of the patent documents that are used as topics.

For the TS task, however, we have seen improvements and we expect even more from domain-specific methods, particularly as we move toward topics involving structure search in 2010 and beyond.

After the experience of the first year, and going into the second year of the campaign, the organizers have a better understanding of how many different ways of thinking about “Chemistry search” exist. The difference between text mining and structure search, both equally valuable in our opinion, is so great that we can hardly point to any group that can do both. As we continue this effort, we should encourage participants to form multi-party teams, to bring together the experience necessary to ultimately provide the most useful tools to practitioners.

At the same time, we encourage practitioners to, first, have patience with us, as this is a basic research effort that will take years to migrate to commercialization and, second, to continuously support such efforts by providing topics (samples of requests for information similar to those they have in their professional life)

and relevance judgments. We acknowledge the wide-spread use of proprietary curated databases. Our efforts to make these available for research purposes have not been successful, and therefore we aim to, in cooperation with practitioners, create tasks which direct research groups toward problems not yet solved by commercial providers, in order to generate new and useful tools.

**Acknowledgements** The authors would like to thank the NIST TREC organizers for supporting this evaluation campaign, Matrixware Information Services GmbH for the patent corpus, Richard Kidd from the Royal Society of Chemistry for providing the initial collection of scientific articles, and all the other editors of the journals that have provided articles in the second year campaign. Last, but certainly not least, the authors express their gratitude to the domain experts who volunteered to provide the manual topics and to evaluate the results of the participants: Teresa Loughbrough, Henk Tomas, Monika Hanelt, Anthony Trippe, Madeleine Marley and her team, and Carlos Faerman.

## References

1. Cetintas S, Si L (2009) Strategies for effective chemical information retrieval. In: Proc of TREC
2. Gobeill J, Teodoro D, Patsche E, Ruch P (2009) Report on the TREC 2009 experiments: Chemical IR track. In: Proc of TREC
3. Gurulingappa H, Müller B, Klinger R, Mevissen HT, Hofmann-Apitius M, Fluck J, Friedrich C (2009) Patent retrieval in chemistry based on semantically tagged named entities. In: Proc. of TREC
4. Hersh W, Voorhees E (2008) TREC genomics special issue overview. *Inf Retr*
5. Hirschman L, Yeh A, Blaschke C, Valencia A (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinf* 6(S1)
6. Jin S, Ye Z, Lin H (2009) DUTIR at TREC 2009: Chemical IR track. In: Proc of TREC
7. Jones KS (1981) Information retrieval experiment. Butterworths, Stoneham
8. Lupu M, Piroi F, Huang J, Zhu J, Tait J (2009) Overview of the TREC chemical IR track. In: Proc of TREC
9. Lupu M, Huang J, Zhu J, Tait J TREC chemical information retrieval—an evaluation effort for chemical IR systems. *World Pat Inf*, to appear
10. Lupu M, Piroi F, Hanbury A (2010) Aspects and analysis of patent test collections. In: Proc of PaIR
11. Mejova Y, Thuc VH, Foster S, Harris C, Arens B, Srinivasan P (2009) TREC blog and TREC chem: a view from the corn fields. In: Proc of TREC
12. Pubmed central. <http://www.ncbi.nlm.nih.gov/pmc/>
13. Soboroff I (2010) Test collection diagnosis and treatment. In: Proc of EVIA
14. Urbain J (2009) TREC chemical IR track 2009: a distributed dimensional indexing model for chemical patent search. In: Proc of TREC
15. Voorhees E, Harman D (eds) (2005) TREC experiment and evaluation in information retrieval. MIT Press, Cambridge
16. Yilmaz E, Kanoulas E, Aslam JA (2008) A simple and efficient sampling method for estimating AP and NDCG. In: SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval. ACM, New York, pp 603–610. <http://doi.acm.org/10.1145/1390334.1390437>
17. Zhao J, Huang X, Ye Z, Zhu J (2009) York University at TREC 2009: Chemical track. In: Proc of TREC
18. Zhao L, Callan J (2009) Formulating simple structured queries using temporal and distributional cues in patents. In: Proc of TREC

# Chapter 6

## Evaluating Real Patent Retrieval Effectiveness

Anthony Trippe and Ian Ruthven

**Abstract** In this chapter we consider the nature of Information Retrieval evaluation for patent searching. We outline the challenges involved in conducting patent searches and the commercial risks inherent in patent searching. We highlight some of the main challenges of reconciling how we evaluate retrieval systems in the laboratory and the needs of patent searchers, concluding with suggestions for the development of more informative evaluation procedures for patent searching.

### 6.1 Introduction

Patent searching is a highly interactive and complex process, often requiring multiple searches, diverse search strategies and careful search management [1]. There are different end-user requirements for different types of patent search and simple performance-based measures of retrieval system functions are often inadequate in expressing the degree to which an Information Retrieval (IR) system might help conduct a successful search.

A particular characteristic of patent searching is the importance of the risk to which a company is exposed if a patent search is poorly conducted. Inadequate tools increase the likelihood of a poor search and increase the level of risk if a company proceeds on the basis of the search.

The claim from most IR evaluations is that measures of recall and precision, implicitly, calculate which system(s) are more likely to reduce this risk by performing more effective retrievals. Therefore, it is argued, we can be more confident about performing a good search with a system that has performed well in system trials. In this chapter we argue that this argument is naïve when considering real operational use.

---

A. Trippe (✉)  
3LP Advisors, Dublin, OH, USA  
e-mail: [tony@trippe.com](mailto:tony@trippe.com)

I. Ruthven  
Department of Computer and Information Sciences, University of Strathclyde, Glasgow,  
G12 8DY, UK  
e-mail: [ir@cis.strath.ac.uk](mailto:ir@cis.strath.ac.uk)

Specifically we consider why recall and precision may give misleading interpretations on system performance, why we need to distinguish the characteristics of different types of patent search and where IR performance variability arises. A core theme in the chapter is the notion of risk: what risks are involved in patent searches, how these connect to measurements of recall and precision and how measurements of recall and precision may misinform rather than enlighten us as to system performance. We conclude with a discussion on how we might increase our confidence in IR system performance as measured in operational environments.

## 6.2 Types of Patent Search

Patent searches go by a variety of different names. Listing the most popular ones you hear terms like: State-of-the-Art, Prior Art, Patentability, Validity, Invalidity, Clearance, Freedom-to-Operate, Novelty and Landscape (see Chap. 1 in this volume). While there may be a large number of terms used to describe patent searches in essence they boil down to four major categories upon which we shall concentrate in this chapter: State-of-the-Art, Freedom-to-operate, Patentability and Validity.

Patent searchers traditionally use these types of descriptions to talk about the searches they perform for various clients whether they are from the legal department or the corporate strategy group. Before formally defining these types of search, it might be useful to think of these various types of searches in terms of the amount of risk they represent to the enterprise. Later we shall compare them to one another on precision and recall scales.

### 6.2.1 Patent Searches and Risk

We define risk as the amount of money that has already been invested in an innovation by an organization pursuing a technological solution to a problem. As the amount of money invested by the enterprise increases the importance of making good decisions about whether to continue funding the innovation and pushing it toward commercialization also increases. With additional funding comes additional risk since the amount of money required to move from one step to the next in taking an innovation to market gets almost exponentially larger.

The pharmaceutical industry provides a perfect example of this concept of increased investment and risk. Early stage projects are expensive in terms of the time spent by the scientific teams in creating new drug entities and having those tested. These are sunk costs and are part of starting a pharmaceutical company in the first place. As new drug entities are discovered, however, decisions need to be made on whether they will be brought forward into what is first called a pre-clinical phase and then a succession of three human clinical trials. Each subsequent stage in this process becomes more expensive than the next as more people are involved in the trials, additional dosing schemes are employed and longer time periods are involved.



As a company approaches a phase III clinical trial the amount of money that will be invested is counted in the hundreds of millions of dollars and pale in comparison to the money that was spent generating a new drug entity and entering it into pre-clinical trials.

Since there is increased risk from substantially increased investment as a new drug entity moves from one stage to the next in the drug discovery process companies have adopted a mantra referred to as “failing faster”. The idea being that if they can find mechanisms for discovering earlier in the process that a new drug entity is going to fail then the company can save themselves a tremendous amount of money by learning as quickly as possible that this is the likely outcome. They cut down on later risk by identifying failure points earlier in the process before larger investments are made.

Analogies can be made to the world of patent searching from this example and many companies follow a similar mantra that if they can discover potential legal impediments to future production earlier in the process then they will save themselves money by changing course based on this knowledge. We can analyze the four major types of patent search by the risk involved.

**State-of-the-Art** This type of search is conducted in order to determine the prevailing technical knowledge in a particular subject area. A practitioner might be entering a new technical area and is interested in learning about the work that has already been done in this space. It is not uncommon for users to be interested in non-patent as well as patent documents in this case since the end goal is to have a thorough understanding of what the current knowledge is in a technical area of interest. People interested in technical or competitive intelligence will also be interested in these types of searches and when they begin to analyze the details of the results they get they will sometimes refer to these as landscaping studies. The sort of details a user can glean from these results are shifts in technology over time, interest in technology sub-categories by company and who the subject matter experts in the field might be. State-of-the-Art searches are typically done at the very beginning of projects before any investment has been made and investigators are trying to determine if an innovation is worth pursuing for a number of reasons. The risk associated with these searches is low and this will have an impact, as we will see later on the corresponding need for precision and recall.

**Patentability** This type of search is usually done in the legal context of determining if a new invention is eligible for patent protection and determining how broadly the claims for the new invention can be written. This type of searches can cover both patent and non-patent literature and are typically looking for references that were published before the filing date of the invention in question. In the United States inventors have up to a year from first public disclosure of an invention to file a patent so some searchers will go back an additional year with their searching to make sure they have found the best references. This is the type of search that will be done by an examiner to determine if they should allow a patent application to be granted.

Even though an examiner will do this search it is important for the applicants to also conduct one since they will often have the time and resources to be more

thorough than the examiner can be. It is also important since knowing the boundaries of the known references will help the attorneys drafting the claims to ask for the broadest coverage possible. Without knowing the scope of the known references it is difficult for the attorney to know how broadly they can write the claims and still expect the examiner to grant a patent.

Patentability searches are done once an inventor has an idea and they have either reduced it to practice or they have a pretty good idea on how they are going to reduce the idea to practice during the preparation of the patent application. Investment has increased since the inventor has spent time discovering the idea and may have used additional time and money reducing it to practice. The total money spent, most of which is fixed costs, is still fairly low and thus the risk involved in this situation while higher than the stage when the State-of-the-Art search was done is still low.

**Freedom-to-Operate** Possibly the most specific type of patent search this particular one is country specific and only applies to in-force granted patents and their claims. A company will ask for a legal opinion on whether a product they are planning on shipping will infringe any existing patents before they launch. There is nothing offensive about this type of search since the interested party is not going to assert patents against anyone else they are simply looking to make sure that they are not going to be infringing someone else's patents. A searcher in this case needs to identify the critical components of the product in question and search country specific claims of in-force patents to see if any of them cover the product components in question. In most cases a great deal of money has gone into a product launch or can be involved with a successful product which is generating a great deal of revenue so it is important for companies to know that they will be reasonably safe from future litigation before they make an even larger investment.

Some companies do Freedom-to-Operate searches reasonably early in the production cycle and follow-up with them frequently to make sure the situation hasn't changed as they get closer and closer to market. These companies are following the "fail faster" philosophy that was mentioned earlier since they recognize that it is better to know about potential legal issues before they make larger investments and involve higher risks. Other companies wait until the trucks are about to leave the warehouses and then conduct a Freedom-to-Operate search as a last item of their checklist before they go to market. At this point a great deal of time, money and effort has gone into the innovation and the amount of investment and risk is pretty high. On more than one occasion companies have trucks filled with product that has been left in a warehouse because a last minute Freedom-to-Operate search has come back with an In-Force patent that could be used against the company later. Regardless of when these searches are applied the risk is much higher than at the Patentability stage and should be considered medium to high.

**Validity** Validity search comprise the largest and most comprehensive of all patent searches. These searches are almost always associated with large sums of money and critical business decisions and as such need to be as comprehensive as possible. This search shares similar characteristics to the Patentability search but is normally far

more comprehensive since there is typically much more at stake when this sort of search is being initiated.

The object of the search is to identify prior art references which will allow a granted patent to be made invalid during a re-examination before the particular patent office of interest or during a court proceeding. Sometimes company will also initiate validity challenges for patents that they are thinking of acquiring especially if they believe these patents will later be used in some type of litigation or another. On the flip side of this a company who is provided with a cease and desist notice will often want to make the patents in question go away by finding invalidating prior art and then entering into re-examination. The prior art references in question can come from the patent or non-patent literature, must be available in the public domain and have to have been published prior to the priority filing date of the patent in question. In the United States there is a one-year grace period on patents filings so some searchers will look back an additional year when they search so they can be sure to avoid this type of situation.

Validity searches are conducted when an organization has received a cease and desist order or are about to spend a significant some of money on a purchase of one sort or another and due diligence needs to be performed in order to justify the transaction. Investment in this case either in the form of production costs and lost sales or in money to be spent on an acquisition are very high and the corresponding risk to the groups making the investment are also extremely high. Since large sums of money are involved and the risk involved is so high companies are willing to increase the resources made available to conduct these types of searches.

Summarizing the searches on our risk continuum we have State-of-the-Art, followed by Patentability; then Freedom-to-Operate, and finally Validity.

The amount of risk involved will have an impact on the resources that are made available to do the searching and in turn this will have an impact on the precision and recall that will be expected in these searches. While risk is not the sole qualifier for precision and recall, there are cases where you have high risk but you do not need high recall per se, it is still useful to keep it in mind as we look at the requirements for these searches.

### **6.2.2 Risk and Recall**

Looking at recall and thinking about a continuum we come across an example where higher risk does not require higher recall. In the case of our highest risk search, Validity, we also find that total recall is not necessarily required. In this type of search it is only necessary to find *one* reference which predates the filing of the patent application in question that describes the invention. In practice most searchers will not stop when they find a single reference and will seek to be as comprehensive as possible but strictly speaking it is not a requirement. Since there is a high risk searchers will often seek higher recall to make sure there are contingencies in place and not rely on a single reference. These considerations put Validity on the low to medium scale with regards to recall.

With Patentability the recall question will depend on who is doing the searching. In the case of an examiner the recall will be the lowest of all the searches we are discussing since they will stop once they find a single reference which will enable them to disallow a claim. They can also take two references and combine them to disallow a claim so they will stop if they find that combination. Patentability searches done by corporate searchers, however, are usually higher in recall since they are helping assist the attorney in deciding how broadly they can write their claims based on how much prior art is out there and how closely (we will do precision next) it matches the invention to be patented. Since the risk is still reasonably low, however, they will not attempt to achieve higher recall since they will reach a point of diminishing returns and making an investment to achieve it would not be economical.

State-of-the-Art searches involve low risk but you would like to achieve a reasonably high recall since the inventor is exploring an unknown area and they will spend time landscaping the area to increase their understanding. Economically speaking recall is sacrificed due to the small investment being made at this point and the bar for diminishing returns is pushed even lower since the expectation is that more comprehensive searching will be done once an actual invention has been discovered and when a product cycle starts.

For recall the top search is Freedom-to-Operate where a single missed patent can come back and be used for a cease and desist action. It is very important to find any and all patents that cover the elements of product to be brought forward to an attorney so they can make a determination as to whether the product will infringe on the patent in question. In order to conduct business not just one patent can be found that an invention may infringe upon but all of them need to be located in order to ensure that the company will not face future legal issues. These searches are referred to as Freedom-to-Operate for this exact reason.

So, looking again at our continuum and comparing recall this time we have Validity and Patentability at the lower end of the scale, State-of-the-Art in the middle and Freedom-to-Operate at the high end. Recall does not correlate with risk necessarily in this comparison with the possible exception of Freedom-to-Operate searches.

### ***6.2.3 Risk and Precision***

Precision maps almost completely to our assessment of risk. State-of-the-Art searches are sometimes called “quick and dirty” since there is not much time invested in doing them and the results often have a large number of false positives contained in them. Also by its very nature this search is exploratory and as such a high degree of precision is not required.

Patentability searches are typically more precise but by their nature are used to explore the boundaries of the prior art so that broader claims can be written to cover more aspects of an invention if warranted so precision is important to cut down on the records that will need to be looked at but not essential. A number of false positives are expected and are part of the process.

Freedom-to-Operate and Validity searches both require a high degree of precision since very specific documents are required in each of these cases. With Freedom-to-Operate the aspects of the produce must be covered in the claims of In-Force patents from the countries of interest. The product must also use all elements of the claimed invention in order to infringe. Finding patents that meet this criterion is a tall order and requires high precision. Similarly, in a Validity search a precise search of the patent and non-patent literature is required to locate references which describe the exact invention covered in a later patent claim either by itself or in combination with another reference.

On the precision continuum we have State-of-the-Art at the low end, followed by Patentability and finally Freedom-to-Operate and Validity.

Looking at each search by its characteristics we can say State-of-the-Art is low risk with low precision and medium recall. Patentability is low risk with low recall and precision. Freedom-to-Operate is high risk requiring both high recall and precision and Validity having the highest risk and requiring high precision but able to get by with lower recall.

Looking at searches in this fashion it is apparent that Freedom-to-Operate searches offer the most difficult challenge for IR researchers. The risks involved are also very high so the expectations will be large and the reluctance to move away from established methods will be severe. Validity is also a difficult task since the risks are so high and the precision requirements so large. State-of-the-Art is where most systems work currently and don't necessarily provide much reward for the effort since they are low risk and are conducted with little in the way of investment. Patentability seems to be the sweet spot for IR research since it offers a reasonable challenge with a good opportunity for return since it is conducted during a stage where resources will be spent to address the issue.

Having outlined the challenges to the patent searcher in conducting a successful search we now discuss some of the challenges IR researchers face in defining appropriate evaluation measures.

### **6.3 Limitations of IR Evaluation**

As in other domains, the evaluation of the retrieval components of patent search systems focuses primarily on laboratory-style evaluation and these evaluations are heavily shaped by the classical models of IR laboratory evaluation. As noted in Carterette and Voorhees (Chap. 3 in this volume), early influential laboratory evaluations included studies such as the Cranfield I and II experiments, SMART evaluation, and the in-depth evaluation and failure analysis of the Medlars search service [2] using small document collections. The experience gained from these studies has been incorporated into the creation of modern test collections where collection size has grown considerably since these early studies. The most widely used test collections come from the Text Retrieval Evaluation Conference (TREC) initiative [3], the

Cross Language Evaluation Forum (CLEF),<sup>1</sup> which are discussed in separate chapters in this volume and NTCIR.<sup>2</sup> The oft-stated value of test collection evaluations are the tightly controlled nature of the evaluation, the statistical rigor with which the evaluation test results can be analyzed and the repeatable nature of the evaluation tests.

The value of IP systems in operational use, however, is influenced by more than the quality of the retrieval system itself and, as has been repeatedly demonstrated in operational tests in other domains, the contextual factors surrounding the *use* of a system (such as organizational concerns, training and experience of the searcher and time available to search) can strongly influence the end results of a search [4, 5]. This gap between real-life practice and laboratory rigor raises three important questions, which we shall examine in the remainder of this section.

1. Are laboratory evaluation measures misleading? Recall and precision are the standard measures for evaluating IR system performance. Although there are many ways in which we can use recall and precision to obtain evaluation measures there are arguments for why they are poor measurements for end-user evaluations unless they are contextualized by other information. In Sect. 6.3.1 we examine some of these arguments and why they raise concerns for determining the confidence we can place in laboratory evaluation performance figures.
2. Are the results of laboratory evaluations sufficiently good at predicting real-life performance? That is, can the results obtained from a laboratory test of an IR system inform us of the potential value of a system in operational environments? In Sect. 6.3.2 we survey some recent work, which indicates a weak correlation between the performance evaluations of systems without user involvement and evaluations of systems operated by end-users.
3. Are laboratory evaluations sufficient? Real-life evaluations incorporate factors that are usually eliminated from laboratory evaluations, such as the expertise of the searchers themselves. In Sect. 6.3.3 we examine some of these factors and outline their importance in reliably measuring system effectiveness.

### ***6.3.1 The Potentially Misleading Effects of Recall and Precision***

Patent search evaluation, similar to other retrieval problems, focuses primarily on recall and precision as measures of system effectiveness. These are long-held measures of retrieval quality and their tight hold on evaluation comes from their intuitive nature: how much of the useful information has my search retrieved (recall) and how much of the information that I have retrieved is useful (precision)? There is also a useful probabilistic interpretation of recall and precision: recall estimating the probability that a relevant document will be retrieved in response to a query and precision estimating the probability that a retrieved document will be relevant [6].

---

<sup>1</sup><http://clef.iei.pi.cnr.it>.

<sup>2</sup><http://research.nii.ac.jp/ntcir>.

Most test collections are constructed using a generally accepted model referred to as the Cranfield model deriving from the Cranfield II tests [7]. A test collection that adheres to the Cranfield model will consist of a set of searchable objects, a set of information requests (or occasionally statements of information problems) and a list of which objects in the collection should be considered relevant for each information request. To ensure fair comparison between systems a number of important assumptions are made. These include the following assumptions.

1. The topics are independent of each other.
2. All objects are assessed for relevance.
3. The judgments are representative of the target user population.
4. Each object is equally important in satisfying the user's information need.
5. The gathering of relevance assessment is independent of any evaluation that will use the assessments.
6. The relevance of one information object is independent of the relevance of any other object.

These assumptions are intended to ensure a fair and accurate comparison between estimates of system performance. The status of these assumptions has shifted over the decades of evaluation research since the original Cranfield model. Assumption 1 is generally adhered to in order to increase the diversity of the test. Assumption 3 is an attempt to ensure external validity of the experiment, i.e. that the results can be generalized to requests beyond those investigated within the test. The level to which this assumption matches most test collections is seriously under-investigated. Assumption 5 attempts to control the internal validity of the study: the assessments used to evaluate the system are not created by the people who designed the study and therefore, it is hoped, that bias will not be introduced into the collection. Assumption 4 is a simplification of real search behavior and many new test collections have graded relevance assessments to allow for more detailed measures of system effectiveness. However, the grades of relevance used often simply reflect amount of relevant material contained within objects rather than quality of relevant material. Assumption 6 is present in most test collections<sup>3</sup> although it is patently false—a system that retrieves duplicates or near-duplicate documents in favor of new and different relevant documents would not be seen as a better system by most users.

Assumption 2 is the assumption that has gathered most attention within the IR evaluation literature, particularly with the rise in test collection size. The early test collections contained small numbers of documents—the Cranfield collection contained only 1400 documents—and it was feasible for exhaustive relevance judgments to be made on the collection. For most collections this is not feasible: it has been estimated that it would take more than nine months to judge an average size TREC collection for a single topic [7]. Not only is this expensive both in terms of time and resources, but over a protracted time period the criteria an assessor will use to judge a document for relevance could change, resulting in inconsistencies in

---

<sup>3</sup>With the possible exception of INEX which does consider the relative relevance of sub-document units which may have overlapping content.

the relevance assessments and therefore in the evaluation results. Indeed, Swanson [8] expressed this as one of his postulates of impotence—statements of what IR cannot achieve—namely, that it is never possible to verify if all relevant documents have been discovered for a request, as one can never examine all documents without unlimited resources while using a strict and static set of criteria for judging relevance. This is, of course, a real challenge for searches such as Freedom-to-Operate searches where the retrieval of all relevant documents is exactly what is required.

The reason that Assumption 2 has gathered so much attention is that exhaustive relevance assessment offers some guarantee that all relevant items have been identified, even if they do not linguistically match the user's query. That is, exhaustive assessments allow the identification of documents that conceptually match the query even if they don't match the user's choice of keywords.<sup>4</sup> Such assessments also allow for deep failure analyses of searches to ascertain why some search topics are more difficult for retrieval systems than others [9]. Such analyses are necessary, particularly with the current trend toward heavy averaging and aggregation of test results over large numbers of topics and collections. Several authors have argued against such approaches, particularly on the grounds that such tests are attempting to prove system hypotheses rather than disproving them. That is, experimenters are trying to prove a system works well rather than attempting to uncover when it will perform poorly. Such tests do not "provide deep insights unless there is some degree of risk in the predictions" [10].

The current model for test collection—the pooling approach—is dependent on queries to create document assessment pools and pooling compensates for exhaustive assessment by the inclusion of diverse systems and manual searching (see Sect. 3.2.2 of Carterette and Voorhees, this volume). The hope is that, if we take sufficient care in sampling the documents to be assessed for relevance, we do not need to exhaustively assess the whole collection. The system-centered evaluation approach, therefore, argues that if we are sufficiently careful in selecting which documents are assessed, and we evaluate on sufficiently large numbers of information requests, then we do not need to assess all documents in a collection.

The nature of test collection construction and, the consequences of Assumption 2, are also important if we consider searching in operational environments. Test collection test results inform us of how well one system performs against another over a set of requests. Many studies have shown the performance of any system across a set of requests is highly variable: systems will perform well for some requests and poorly for another. What IR tests cannot predict is how well a system will perform for a given request. This means, in operational environments, that the *searcher* must decide how well the system is performing for any given request. In many search situations such variability might not matter, in patent searching it is more difficult to accept that some requests will be handled well and others not.

Blair and Maron [6] in one of most famous IR evaluation studies demonstrated that even experienced searchers can radically underestimate the proportion of relevant material obtained from an interactive search and that the quality of the

---

<sup>4</sup>Exhaustive query assessments also mean that we can assess the quality of the original query itself.



searcher's queries can affect the *perception* of system performance. Although we can form intuitions about whether a system is returning relevant material we cannot assess, simply based on the retrieved results, how much relevant material has been returned or how much remains to be retrieved. Blair and Maron in [6], and later [10], proposed four main reasons for the findings from their study:

1. Users often cannot predict which words are good at retrieving relevant material. In spite of detailed knowledge about the material with which they were involved, the searchers in their study could not identify useful search terms to retrieve important subsections of the database. However, they could consistently recognize useful information when it was presented to them. Common problems with querying included lack of knowledge of synonyms used in the unretrieved relevant material, poor handling of spelling mistakes relating to important terms, and other oft-seen dilemmas in creating search requests.
2. The large size of the document collection meant that attempts to control precision—and hence make the results sets manageable—reduced the recall of searches. However, this resulting in the elimination of important relevant material from the search results.
3. Searchers can mistake document retrieval for data retrieval. That is they describe the data they want to retrieve rather than the content of the documents they want to retrieve.
4. Overestimations of recall in laboratory tests give a false sense of security. In [10] Blair pointed out that poor laboratory tests can artificially inflate recall estimates. As noted above, test collection creators compensate for lack of exhaustive assessment by increasing the diversity of systems used to supply documents for assessments. The hope is that such diversity will lead to representative relevant documents being found. If the diversity is weak then the recall figures can be artificially inflated because the relevant documents may be easier to find. Knowledge that one is using a good system can also give the searcher the perception that they are finding more of the relevant documents than they actually are.

What Blair showed was that, even by submitting variations of query terms adjusted through trial and error, as in a typical search session, the likelihood of a searcher finding a substantial proportion of relevant documents can be low, a finding that has been verified across a number of studies [10]. An explanation for this limitation is that the intellectual content of a document is difficult to represent automatically: a document can be about a topic without ever mentioning key terms or phrases that a user may expect to appear. In addition, the query terms chosen by the user may not discriminate between relevant and non-relevant documents, especially as the collection size grows [11]. A user searching for documents on a new subject may not select terms that are representative of the subject they are searching *and* that discriminate such documents from the non-relevant documents which share similar vocabulary. Consequently, not all potentially relevant documents will be retrieved through keyword matching techniques alone.

In a real search situation, a search can only estimate what is hidden (the unretrieved relevant documents) by what they have already found and by the quality of

their attempts to find these documents. In [12] Blair argues that the latter is difficult to measure and searchers are often forced into intuitive reasoning about search strategies. One particular process, known as ‘anchoring’ is of particular interest in searching. Anchoring is a psychological process in which people estimate unknown values (the quality of queries in our case) by starting from an initial value which “may be suggested by a formulation of the problem”. If a particular query is seen as good, either because it retrieves relevant documents or the searcher believes it to consist of good indexing terms, then they will retain, and modify the query, rather than attempt new queries, ones which may be better at retrieving different types of relevant material.

Blair and Maron’s final point is also an important one for real search situations where the effort involved in conducting a search must be balanced against the cost of conducting a search: finding a number of relevant documents is not a sole indicator of good retrieval performance, as the proportion of relevant documents *missed* is not known unless it is quantified through other means. Swanson refers to this as the “fallacy of abundance”—discovering a (substantial) number of documents about a request creates an illusion that little remains hidden [8]. Good precision, in particular, can give the false impression that the system has good recall.

There are two issues relevant for patent retrieval. Firstly, the degree to which recall and precision as measured in laboratory tests are actually informative of the likely performance in real situations. In the most challenging patent searches simple measures of recall and precision may have little predictive power because what reduces company risk is not simply the ability to find relevant material but to have performed a comprehensive search. Very few system evaluations tackle the issue of how dependent system performance is on the initial request or how variable is the system’s performance. Therefore, the end-user’s own expertise and judgment plays a large role in the system’s overall performance. Secondly, and a consequence of the above discussion, we need to investigate the end-user’s abilities to make judgments about recall and precision in operational environments. Blair and Maron’s studies indicated potential pit-falls about making such decisions in real-life settings, particularly when cost and time must be balanced against effort. As we will discuss in Sect. 6.4, there are ways in which we can estimate the skill of the person operating the system.

### ***6.3.2 Predicting Performance from Laboratory Tests***

One of the core claims for test collections, as noted for example in Sanderson and Zobel [13], is that the relative performance of systems from a test collection evaluation tells us something about how the systems will perform in operational settings. This is trivially true in extreme cases; a system that continually retrieves the wrong documents in a controlled test collection evaluation is unlikely to perform well in an operational setting. The test collection approach, typically but not always, concentrates on single retrieval runs. Some authors, such as Spärck Jones [14], have argued

that this is not an issue; systems that perform well on one retrieval run will perform well in most retrieval situations and performance on single retrieval runs give us an indication of how well a system will perform iteratively. However, single-run evaluation limits our ability to evaluate the effect of known aspects of how humans assess relevance, in particular dynamic effects such as the development of relevance criteria across a search [15] or the effect of the order of assessment [16].

However, the general claim that single-run retrievals are good estimates of overall system performance has not been convincingly demonstrated so far, partly due to the few comparisons in operational settings and, partly due to the impact that user adaptation and interfaces have on the level of retrieval effectiveness of a complete system. What has been investigated is the degree to which laboratory tests and user tests align. This is *not* the same as tests in operational environments, where many contextual factors will intervene.

Hersh et al. [17], who were one of the first authors to try direct comparisons between test collection and interactive experiments, show that results from a test collection do not necessarily follow to the interactive case because the interactive aspects of a system can interfere with the results. Their investigation also raises the question of what are *meaningful* differences between retrieval results: how much better does one system have to be over another in a test collection evaluation for us to be convinced that it is indeed a better system and are these differences ones that are observable to users of the systems? Since Hersh and Turpin's paper there have been a large number of attempts to shed light on the second question. The evidence is distinctly mixed. Kelly et al. [18] for example, showed that end-users could distinguish detect differences in retrieval performance but within tightly controlled environments where the users were forced to interact in specific ways. Hersh and Turpin's later results and Smith and Kantor's very robust study indicated, however, that users can compensate for the performance of poor systems [19] and, to a degree, undo the effect of good systems by raising their threshold for relevance [20].

Harter, [21], for example, criticized the standard test collection model of evaluation because it ignored the variation in why relevance assessments are made for specific information requests. Relevance assessments in operational settings are heavily contextualized by the situation in which the assessments were made, and this context includes the person making the assessment.

Spärck Jones, in a later paper, also mentioned the importance of context and notes (of TREC in particular) "context is not embraced, but reluctantly and minimally acknowledged, like an awkward and difficult child. This applies even where explicit attempts have been made to include users (real or surrogate)" [22]. Limited attempts to incorporate context within test collection environments have been attempted, notably in the TREC Hard and CiQA tracks, but these have typically related to the contextual information within the query rather than contextual factors which might affect operational use of a system.

### 6.3.3 *Are Laboratory Evaluations Sufficient?*

Few evaluation measures, and not those typically associated with test collections would take into account other factors that are important to users such as the validity of information, the ability of a searcher to understand the information retrieved, the source of the information or the searcher's prior knowledge about a search topic [23]. Many studies (such as [24, 25]) have shown that even for expert searchers their confidence or prior knowledge in a search topic can affect their assessments of a document's relevance: they will mark different documents as relevant, and different numbers of documents, independently of how those documents were retrieved. Voorhees, in a tightly controlled study, estimated the difference in opinion between assessors as around 35%; Ruthven et al. [24, 25] indicated that differences also occur with individual assessors depending on their prior relationship to the search topic. Further, as noted above in Sect. 6.3.1, a searcher's behavior can strengthen the performance of a poor system or weaken the performance of a good system.

The question then arises as to what degree measures such as recall and precision obtained from laboratory studies actually help predict how good a patent search might be? If relevance assessments change depending on who is doing the assessment, then how much confidence can we have in evaluation measures based on relevance: if a different patent searcher conducted the same search would we have different results? In operational environments, especially for searches with high risk, patent searchers can interact with each other to minimize the possible negative effects of individual variation in relevance judgments and search strategies.

However, as noted in Sect. 6.3.1, this places the emphasis for success onto the searcher and away from the system. A good set of evaluation measures would recognize and reward systems that offer support for end-users in making challenging search decisions. The patent searches outlined in Sect. 6.2 are not simple searches; they are active processes where the end-user must engage in a process of sense-making—understanding and interacting within information in complex ways to make a decision or recommendation. What makes a good IR system for this type of search behavior is the ability of the system to make better sense of the search results and have more confidence in the accuracy of the outcome. This cannot be measured simply by performance evaluation but requires evaluating the process of searching. So how can we estimate the value of an IR system in helping successfully conduct a patent search?

In Sect. 6.4 we try to address this final question, building on the discussion in the previous sections, by outlining how we can gauge levels of trust in various parts of the IR process.

## 6.4 Evaluating Real Patent Retrieval Effectiveness

Any evaluation measure, implicitly or explicitly, carries a definition of success. This definition of what it means to succeed in an evaluation carries with it, in turn, the

definition of what we see as the task of IR systems. In this chapter we argue that the role of IR systems is to reduce overall risk; partly this is associated with measures of recall and precision (although simple measures may be too blunt) but the highly intellectual and interactive role of the patent search system (as a whole) needs to be incorporated into the evaluation.

One way of viewing IR evaluation is as a series of evaluation layers, each with distinct methodologies, metrics and questions. Lower evaluation levels comprise highly constrained, specific investigations on single system features; higher levels contain broader multi-faceted investigations on the searcher *and* system. At the lower levels, for example, evaluations are typically on the algorithmic properties of system components and are run as performance tests conducted without human involvement. Higher levels will examine the interactive nature of the system to consider the degree to which the whole system supports an end-user's information search. Appropriate metrics here will include both measures of the search products and the process of searching [26]. Product metrics, those that measure the end results of searching, may include aspects such as the number of relevant documents found, search satisfaction or time taken to complete a search. Process measures, on the other hand, consider how these products arose within a search and could include factors such as the ease of completing a search, the user understanding of the interface functionality, their increase in confidence in using the system and use of system features.

As noted in Sect. 6.3, there are major differences between algorithmic evaluations and operational trials.

1. The effectiveness of a real patent search is dependent on use of multiple systems and the searchers' ability to use them. Sections 6.3.2 and 6.3.3 outlined some of the reasons why IR evaluations may not give us good predictions of how well a system performs in operational tests.
2. IR evaluation is based on generalizations. As noted in Sect. 6.3.1 IR evaluations tell us which systems are better for an average request. However, their performance across topics is very variable.
3. Individual estimates of recall and precision are affected by individual variation in how a searcher assesses relevance and what is returned by the system. It is far easier to reason about what is returned by a system than to reason about what is not returned.

Patent searching is a complex form of searching and one that involves multiple searches, collaboration with other people and heavy use of instinct and experience. So what types of evaluation are useful in understanding the success of an IR system for different types of patent searching? Arguably the success of any IR system is how well it supports the user in an information task and measuring this will involve a number of different measures some of which will be product based and some will be process based. However, as noted in Sect. 6.3, the ultimate purpose of IR tools within the IP process is to reduce risk by helping end-users discover the required information or, alternatively, be reassured that certain information does not exist. Current laboratory evaluation measures do not help assess the degree to which an IR

system has helped reduce this risk. Due to the variability in IR system performance, a user cannot guarantee any minimum level of performance for an *individual* search request. Nor can system designers assert, concretely, what level of confidence they should have in individual system components reducing risk because, as noted in Sect. 6.3, risk and recall/precision are not linearly related.

What we can try to develop are evaluation approaches that help estimate the confidence we should have in different system components. That is, how might we estimate what levels of trust we can have in parts of the retrieval process? If we have low levels of trust, then the end-user needs to do additional work to compensate for lack of system performance.

### ***6.4.1 Product Based Measures for Evaluating Real Retrieval Effectiveness***

Product measures are common in IR. Recall and precision can be used flexibly to give different estimates of system performance and different estimates are useful for different purposes. For a State-of-the-Art search reasonable recall is required, low precision perhaps tolerated but debatably diversity of results is more important. Systems that artificially boost recall at the expense of missing important sections of the recall base could give the false impression that higher recall has been achieved. Systems may also be rewarded for retrieving some types of documents over others. In landscaping studies it may be more useful for a searcher to have overview documents than narrowly focused documents. Calculating recall and precision over different document sets could be useful here.

For Validity searches very precise results are required. Unlike State-of-the-Art searches where we know there is material to be found but not sure what form it may take, in Validity searches the question is whether the material is there to be found. In such a case a useful evaluation metric may be final user confidence in the results of their search. A system that has a very high degree of topic variability (some queries are very successful, others very unsuccessful) offers little confidence in the performance on a new search. In such a situation the searcher may have to expend more resources, time and cognitive, to complete the search but with little guidance from the system as to how effective the search has been.

Product based metrics often focus on different systems with the same request; what they often fail to do is determine the variability of different requests on the same system. A useful product based metric, particularly in light of the discussion in Sect. 6.3.1, is how variable is system performance to the query formulation. High variation, particularly for best match systems, offers little confidence in the overall system performance and, again, increases the effort the searcher must expend on the search.

### 6.4.2 *Process Based Measure for Evaluating Real Retrieval Effectiveness*

Process based measures are useful for identifying the factors that lead to success and involve analyzing the stages that lead to the end products of a search. In particular, for complex tasks where searchers may spend long periods of time on each search, process metrics are useful for identifying which search decisions are critical and which decisions need different types of system support.

Process measures are often difficult to develop and are subject to variation within the user population. However, process models can be used to (a) understand the processes of searching and (b) analyze success factors within each stage. An example of the latter is the University of Tampere's Query Performance Analyser [27], a tool for assessing how good a searcher is at the task of creating search requests. Such tools can help identify the relative contribution of the person conducting the search but also the contribution of the system to a successful outcome. Such knowledge could increase our confidence in the results of a search (in the case of high user and system abilities) or estimate what level of doubt we should retain after conducting a search. Understanding the process of searching within a professional domain like patent searching can also uncover the major sources of variation within patent searches and move toward correcting the sources of variation. Many disciplines use such process models to increase confidence in the overall process of completing tasks.

For high risk tasks, such as Freedom-to-Operate, which requires both high recall and high precision, we could ask how individual searchers balance these requirements by the choice of search strategies and whether some strategies are more effective than others. Thus we can hope to move toward a more formal evaluation strategy for patent searching.

## 6.5 Conclusion

This chapter considers evaluating *real* retrieval effectiveness; retrieval effectiveness within an operational setting rather than in a controlled laboratory setting common to most IR evaluations. Deciding what to measure in evaluation is a crucial decision. It is worth reiterating the general point that any evaluation approach tends to distort what it tries to evaluate. Evaluation as an activity highlights some aspects of the phenomenon being studied and ignores others. As Hersh and Turpin [20] demonstrated, employing simple relevance metrics in user evaluations can give misleading results because simple metrics may ignore the factors that influence decisions. In this chapter we have argued that retrieval system evaluation needs to provide a richer and more realistic account of the role of systems in reducing risk.

Each domain has its own challenges and presents new challenges to IR. IR researchers typically look at precision and recall simultaneously and measure their methods by how techniques stack up against both elements. When it comes to patent

searching it might be more productive to separate these functions so that they can be maximized independently. It has been demonstrated that risk, precision and recall do not follow the same linear path when discussing the various types of patent searches. Since this is the case it might be more productive to begin with creating methods that produce high recall exclusive of precision. Once this is accomplished the results can be ranked using different methods to improve precision and manage the way the results are shared with the searcher. It will likely be the case that different methods will be used to provide higher recall than those that can be employed to share records with higher precision. Instead of expecting a single method to do both it would be useful to the patent searching community if the process was done stepwise to maximize the value to the user.

It is received wisdom in the IR community that the variation between search requests is the greatest source of variation in retrieval system performance and such variation is greater than the variation between end-users. However, such claims are based on relatively artificial settings and we still have relatively little empirical evidence on what components of a retrieval system are actually useful and the relative contributions of searcher and search system to overall success in patent searching.

We have, albeit briefly, suggested some evaluation directions that may help identify fruitful research directions in patent search evaluation. There are considerable challenges, particularly around issues of confidentiality, to be tackled, but if we are to move toward better evaluation procedures, then we need to be able to ask basic questions about the processes and decisions involved in operational patent environments.

## References

1. Joho H, Azzopardi L, Vanderbauwhede W (2010) A survey of patent users: an analysis of tasks, behavior, search functionality and system requirements. In: 3rd symposium on information interaction in context (IIIX '10)
2. Spärck Jones K, Willett P (eds) (1997) Readings in information retrieval. Morgan Kaufmann, San Francisco
3. Voorhees EM, Harman D (eds) (2005) TREC: Experiment and evaluation in information retrieval. MIT Press, Cambridge
4. Ingwersen P, Järvelin K (2005) The Turn: Integration of information seeking and retrieval in context. Springer, Heidelberg
5. Hansen P, Järvelin K (2005) Collaborative information retrieval in an information-intensive domain. *Inf Process Manag* 41:1101–1119
6. Blair DC, Maron ME (1985) An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Commun ACM* 28:289–299
7. Voorhees EM (2002) The philosophy of information retrieval evaluation. In: CLEF '01: Revised papers from the second workshop of the cross-language evaluation forum on evaluation of cross-language information retrieval systems, pp 355–370
8. Swanson DR (1989) Historical note: Information retrieval and the future of an illusion. *J Am Soc Inf Sci Technol* 39:92–98
9. Voorhees EM (2005) The TREC robust retrieval track. *ACM SIGIR Forum* 39:11–20
10. Blair DC (1996) STAIRS redux: Thoughts on the STAIRS evaluation, ten years after. *J Am Soc Inf Sci Technol* 47:4–22



11. Blair DC (2002) The challenge of commercial document retrieval, Part I: major issues, and a framework based on search exhaustivity, determinacy of representation and document collection size. *Inf Process Manag* 38:273–291
12. Blair DC (1980) Searching biases in large interactive document retrieval systems. *J Am Soc Inf Sci* 31:271–277
13. Sanderson M, Zobel Z (2005) Information retrieval system evaluation: effort, sensitivity, and reliability. In: 28th annual international ACM SIGIR conference on research and development in information retrieval, pp 161–169
14. Spärck Jones K (2005) Epilogue: Metareflections on TREC. In: Voorhees EM, Harman DK (eds) *TREC: Experiment and evaluation in information retrieval*. MIT Press, Cambridge, pp 421–448
15. Vakkari P (2000) Cognition and changes of search terms and tactics during task performance: a longitudinal study. In: *RIAO 2004 (Recherche d'information assistée par ordinateur)*, pp 894–907
16. Huang MH, Wang HY (2004) The influence of document presentation order and number of documents judged on users' judgements of relevance. *J Am Soc Inf Sci Technol* 55:970–979
17. Hersh WR, Turpin A, Price S, Chan B, Kraemer D, Sacherek L, Olson D (2000) Do batch and user evaluation give the same results. In: 23rd annual international ACM SIGIR conference on research and development in information retrieval, pp 17–24
18. Kelly D, Fu X, Shah C (2010) Effects of position and number of relevant documents retrieved on users' evaluations of system performance. *ACM Trans Inf Syst* 28:9:1–9:26
19. Smith CL, Kantor PB (2008) User adaptation: good results from poor systems. In: 31st annual international ACM SIGIR conference on research and development in information retrieval, pp 147–154
20. Hersh W, Turpin A (2001) Why batch and user evaluations do not give the same results. In: 24th annual international ACM SIGIR conference on research and development in information retrieval, pp 225–231
21. Harter SP (1996) Variations in relevance assessments and the measurement of retrieval effectiveness. *J Am Soc Inf Sci Technol* 47:37–49
22. Spärck Jones K (2006) What's the value of TREC—is there a gap to jump or a chasm to bridge? *ACM SIGIR Forum* 40:10–20
23. Barry CL, Schamber L (1998) Users' criteria for relevance evaluation: a cross-situational comparison. *Inf Process Manag* 34:291–236
24. Ruthven I, Baillie M, Elswailer D (2007) The relative effects of knowledge, interest and confidence in assessing relevance. *J Doc* 63:482–504
25. Ruthven I, Baillie M, Azzopardi L, Bierig R, Nicol E, Sweeney S, Yakici M (2008) Contextual factors affecting the utility of surrogates within exploratory search. *Inf Process Manag* 44:437–462
26. Borgman CL, Hirsh SG, Hiller J (1996) Rethinking online monitoring methods for information retrieval systems: from search product to search process. *J Am Soc Inf Sci Technol* 47:568–583
27. Sormunen E, Pennanen S (2004) The challenge of automated tutoring in web-based learning environments for IR instruction. *Inf Res* 9:169

## Part III

# High Recall Search

One of the problems of patent search is that a single relevant missed patent or other document can invalidate an otherwise sound patent. This is why patent searchers often say that they require 100% recall (the highest level possible): that is, they require the search system to guarantee to return absolutely all relevant documents, no matter where (USPTO, web, film archives), in what form (XML, PDF, paper), or language (English, French, German, Indonesian, Welsh). In practice, no real system delivers 100% recall all the time, and often searchers actually do not want 100% recall—they will get too many, redundant relevant documents. What they really want is a balance between precision (the proportion of relevant to irrelevant documents in the results list) and recall, which gives a strong guarantee that highly relevant documents are returned high in the results list. If no highly relevant documents are returned, there is a high probability that there really are no relevant documents.

As we have seen in the introductory chapters, a patent search task always involves a phrasing such as ‘find all documents which...’. From the evaluation part, we know that technically, this is easily done: returning to the user the entire collection being searched on will inevitably also contain all relevant documents within that collection. The patent search task, in addition to the explicit requirement of finding all relevant documents, has a series of implicit requirements: ‘find only the relevant documents’ and ‘find them in a reasonable time span’. Therefore, when we talk about High Recall search, we must always keep in mind that at the same time we are talking about High Precision search. However, what exactly ‘high’ means in either of those contexts is still ambiguous. Perhaps a better term would be ‘satisfactory for the task at hand’, but that would not be very marketable.

The challenge for patent information retrieval here is to find mechanisms that will give patent searchers search result sets which they can effectively use for the task at hand. Furthermore, to find means for some cases, like patentability or invalidity searches, that allow the “golden nugget” of the novelty breaking document to be found quickly and easily, while giving the searcher a high degree of confidence that nothing relevant has been missed, if no golden nugget can be found.

The part starts with a very fundamental question, asked by Bache: ‘Can we really find all the relevant documents?’. The problem is very fundamental: given an indexing system, is it possible that some documents become inaccessible because of the way the system works? Or, more generally, how fair is a system in its representation

of the documents indexed and how does this fairness impact the expected success rate of our queries?

The part continues in Chap. 8 with Tomlinson and Hedin taking a look at efforts done to understand high recall in a similar domain: e-discovery. Based on the TREC Legal Track, we see that contemporary search technologies can outperform Boolean models, but also learn of the many pitfalls of evaluating systems expected to return a large number of documents. The authors discuss ways to estimate recall, precision and the F1 measure based on a manageable amount of human evaluation, and point out the extreme effects that human assessment error can have on these estimations.

The next chapter by Azzam and colleagues is of a more technical nature and principally addresses scalability issues—how we take promising techniques for improving recall, which work well on small collections as were used in the past, and adapt them to operate on today and tomorrow’s terabyte and petabyte collections. They also cover methods to improve the precision of the search, which is also a focus of Parapatics and Dittenbach in the following chapter, in which they look at methods to improve the processing of claims, and especially dealing with the complexity of their language, which often compromises the effectiveness of traditional bag-of-words IR techniques.

Finally, the part concludes by addressing the question of what we do after we retrieve a large set of potentially relevant documents. In that sense, Koch presents a system to move away from the list of documents towards a more intuitive graphical display, which can operate with very large sets of documents.

# Chapter 7

## Measuring and Improving Access to the Corpus

Richard Bache

**Abstract** Retrievability is a measure of access that quantifies how easily documents can be found using a retrieval system. Such a measure is of particular interest within the patent domain, because if a retrieval system makes some patents hard to find, then patent searchers will have a difficult time retrieving these patents. This may mean that a patent searcher could miss important and relevant patents because of the retrieval system. In this chapter, we describe measures of retrievability and how they can be applied to measure the overall access to a collection given a retrieval system. We then identify three features of best-match retrieval models that are hypothesised to lead to an improvement in access to all documents in the collection: sensitivity to term frequency, length normalization and convexity. Since patent searchers tend to favour Boolean models over best-match models, hybrid retrieval models are proposed that incorporate these features while preserving the desirable aspects of the traditional Boolean model. An empirical study conducted on four large patent corpora demonstrates that these hybrid models provide better access to the corpus of patents than the traditional Boolean model.

### 7.1 Introduction

A key feature of patent searches is that they are recall-dominated—there is a significant cost associated with failing to retrieve relevant documents [13]. This contrasts with many everyday search tasks such as an Internet search where the proportion of retrieved documents considered relevant to the user (precision) is considered the most important aspect of effectiveness of the information retrieval (IR) system in question. Although precision can be measured quite easily, by performing relevance judgements on the retrieved items, recall cannot be measured practically for all but the smallest of corpora. Recall measurement requires performing relevance judgements on every document and this is not feasible for the size of corpora used in patent search.

---

R. Bache (✉)

Department of Computer and Information Sciences, University of Strathclyde, Glasgow G4 1XH, Scotland, UK

e-mail: [richard.bache@gmail.com](mailto:richard.bache@gmail.com)

An alternative approach has been to measure the degree of access that an IR system affords over a given (typically large) corpus. A recently proposed measure of access, *retrievability* [3–5, 9], is therefore an attribute of particular interest in the patent domain. Essentially, the retrievability of a document is the ease with which that document can be retrieved; document retrievability depends upon the patent collection and the IR system used. A document with low retrievability is likely to be very difficult, if not impossible to find, while a document with high retrievability is likely to be much easier to find. For a given corpus, different IR systems will yield different levels of retrievability across the population of documents. It is therefore important to select an IR system that ensures that all documents are as accessible as possible. This is particularly the case in patent retrieval; because if patent searchers employ IR systems that limit their ability to retrieve particular documents in the collection, this could mean missing relevant documents. Early work using retrievability has provided interesting insights into the problem of documents accessibility [3]. In [5] it was shown that different best-match retrieval systems provided substantially different levels of retrievability across a document collection, while in [8, 9] it was shown that pseudo-relevance feedback can skew the retrievability of documents (i.e. some documents become much more retrievable than others). With such variations in access to documents arising due to the retrieval system, it is important to quantify and understand its influence on the retrieval of documents.

In this chapter, we consider the influence of different retrieval systems on large patent corpora, and determine the retrievability of patents when using such systems. The overall access to the patent corpora is determined to provide an indication of how accessible the population of documents is for each given system. Since patent search is often conducted using traditional Boolean systems (i.e. exact-match retrieval models), we shall examine these types of retrieval models, and compare them against best-match retrieval models. Best-match models have been favoured in the IR research community because they have been shown to deliver excellent retrieval performance. However, within the patent domain, searchers prefer exact-match models because of custom and tradition, the precise interpretation of Boolean queries, and the legal and regulatory requirements that are often imposed. To improve the access of exact-match models, while preserving these required features, we also consider a series of hybrid retrieval models. These models accept a Boolean query and provide a crisp cut-off between retrieved and non-retrieved documents, as the traditional Boolean model already does, but they incorporate a number of features of best-match models that improve the access to the collection. An empirical study using the MAREC patent test collection builds on a previous pilot study [7] and provides evidence that these features often improve access (i.e. they make access to individual documents more equal). It is shown that when these features are incorporated within the hybrid models this leads to improved access over the traditional Boolean model.

The rest of the chapter is organized as follows. In Sect. 7.2, we summarize the reasons why users in the patent domain prefer Boolean queries and, by implication, models which accept queries in this form. Section 7.3 formally defines the concept of retrievability and describes how it can be measured. Such measurement requires a

large number of representative queries, so in Sect. 7.4, we consider both the quality and quantity of queries. Section 7.5 identifies and explains the three features that were hypothesized to increase access. In Sect. 7.6 we give formal definitions of the variants of Tf-Idf and BM25 used for this study and then define the hybrid models. Then, Sect. 7.7 presents the results of the empirical study we conducted to analyze these models on a number of different patent corpora. Finally, Sect. 7.8 concludes with a summary of findings.

## 7.2 Patent Searching

A patent searcher typically requires accessing large corpora, consisting of millions of documents, in order to perform a variety of search tasks [13]. By considering some of the most common search tasks, we can see that the ability to access every potentially relevant document is key to the likely success of each task:

**Novelty Search** given a patent application, the search task is to ensure that the claims made for the new invention have not been previously patented or documented.

**Validity/Invalidity Search** The search task is to investigate existing patents to determine whether their claims are enforceable, or to determine whether any other patents violate an existing or currently held patent.

**Freedom to Operate** A search is instigated to determine if a proposed course of action violates an existing patent.

To accomplish these tasks, patent searchers often prefer exact-match models where the query is submitted in a Boolean form using the AND, OR and NOT operators [6, 10]. In response to such a query, a system employing an exact-match model, will return all the documents for which the query is true. Since the documents returned are not ranked they are usually presented by some ordering criterion such as date.

The approach taken by patent searchers contrasts with many other areas of IR research where queries are submitted as unstructured lists of words and best-match models are used to rank the documents. Such best-match models have the advantage that they can take into account not only the presence or absence of a query term in the document, but also its frequency. It is perhaps for this reason that best-match models have found favour within the IR community as these often result in significantly better retrieval performance (in terms of precision and recall). Nevertheless, the Boolean exact-match model remains popular in the patent domain and this is partly due to the nature of the searches that take place [14, 22]. The usage of the exact-match models stems from the following reasons:

**Custom and Practice** Practitioners have been trained and are used to formulating Boolean queries. The habit of always performing such queries may make them less likely to change, especially if their current practice is effective in finding the required documents.

**Extra Information Content** For a given number of query terms, the addition of Boolean operators and brackets adds more information to the queries. Thus very precise queries can be formulated which have a clear interpretation.<sup>1</sup>

**Demonstrating Due Diligence** The fact that there is a crisp cut-off means that a patent searcher does not have to make an arbitrary decision where to stop examining the documents. This protects him/her against the accusation that ‘if they had only looked a little further they would have found the document in question.’

**Model Intuitiveness** Extending a query using either the AND or NOT operator will retrieve fewer documents. This contrasts with best-match models where adding an extra query term will retrieve the same number or more documents. Patent searchers carefully fashion a query specifically to reduce the number of retrieved documents to make the exhaustive viewing of each document feasible. However, the NOT operator is known to be problematic since it may lead to the exclusion of a document which was relevant but addressed other topics as well.

Therefore any new model for patent search need to provide the same functionality of exact-match models, or at least handle AND and OR operators, but preferably also the NOT operator. Also, it must provide a crisp cut-off between the retrieved and non-retrieved. In this work, we examine hybrid models which combine features of exact-match models with features that improve the access within best-match models, in an attempt to obtain the best of both worlds (i.e. exact-match models with improved access).

## 7.3 Retrievability

The accessibility of information in a collection given a retrieval system has been considered from two points of view, the system side i.e. *retrievability* [5] and the user side *findability* [15]. Retrievability measures provide an indication of how easily a document could be retrieved using a given IR system, while findability measures provide an indication of how easily a document can be found by a user with the IR system. Here we consider the access based measure of retrievability (see [5] for more details and [2, 7–9] for examples of its usage in practice).

### 7.3.1 Definition of Retrievability

The general formula for the retrievability measure of a single document given in [5] (with modified notation) is

$$R(d) = \sum_{q \in Q} p(q) \cdot f(\delta(q, d), \theta), \quad (7.1)$$

---

<sup>1</sup>Paradoxically the output of such a model is either 1 or 0 and this contains less information than the real number yielded by best-match models.

where  $Q$  is the set of all possible queries,  $p(q)$  denotes the probability of query  $q$  being used,  $f(\delta(q, d), \theta)$  is the utility function (note that a high value is good) with  $\theta$  as a parameter and  $\delta(q, d)$  a measure of the cost involved in accessing  $d$  given  $q$  (i.e. going down the through the ranked list of documents incurs a cost). It is not possible to create an exhaustive list of queries, so a subset  $Q' \subset Q$  is created to form an estimate. Since this subset is usually generated artificially and the queries are not based on any actual information need, we are not able to assign a likelihood to each query other than by assuming  $p(q) = \frac{1}{|Q'|}$  is the same for all queries. This then becomes multiplication by a constant. Since we are only interested in this measure relative to other documents, it can be ignored. Thus to provide an estimate  $\hat{R}(d)$  of document retrievability we write:

$$\hat{R}(d) = \sum_{q \in Q'} f(\delta(q, d), \theta). \quad (7.2)$$

To represent the diversity of possible queries that a user could submit, any empirical study will require a very large sample of such queries. It is for this reason that the queries are generated automatically (see Sect. 7.4 for details).

### 7.3.2 Utility Functions

It is assumed that when presented with an ordered list of retrieved documents, the user will start at the top and work their way down. Therefore within IR, the measure of distance  $\delta(q, d)$  is the rank of the document. Given that a patent searcher will choose how many documents to examine, we shall use a *cumulative* measure of retrievability where the utility function gives a score of 1 to the top  $\tau$  ranked documents, and 0 otherwise. In our experiments we shall take measurements of retrievability at five cut-offs, where  $\tau = 10, 20, 50, 100$  and  $200$ . Note that, on average, a patent searcher examines around 100–200 documents per query [14, 22].

### 7.3.3 A Measure of Collection Access

Once we have estimated the retrievability of all the documents in a collection, we wish to calculate some overall measure of access for the collection given the IR system and corpora. Since the retrieval of one document at a particular rank means that another document cannot be retrieved at the same rank, then documents compete to be retrieved, i.e. if one document appears in the top ten retrieved items then, by definition it will displace another document. Indeed, documents become less retrievable precisely because others become more retrievable. What is of interest here is the distribution of retrievability over the population of documents and whether the retrieval system provides a similar amount of retrievability to each document in the collection (or provide a degree of equality to all documents). For example, we



can imagine that we have a retrieval system which only retrieves one particular document regardless of the query. This document would have a very high retrievability score, but all the other documents would have 0 retrievability. Since we would like to ensure that patent searchers can access all documents as easily as possible, then making all documents as equally retrievable as possible would improve access to all parts of the collection. Essentially, we would like the retrieval system to afford all documents with similar retrievability. However, due to the characteristics of the documents this might not always be possible—though the aim is to strive for equality.

In economics, there is a standard method of measuring wealth and income distributions to determine the level of equality. Here, we apply this method to the measures of document retrievability. The Lorenz curve [12] provides a graphical representation of the distribution of individual retrievability scores. The documents are ordered by increasing value of their respective score and then the cumulative score is calculated. This is plotted against the cumulative number of documents. Figure 7.1 gives an example; note that both axes have been normalized. The unbroken line shows complete equality—i.e. all documents are equally retrievable. The heavy, dashed line shows total inequality, only one document is ever retrieved no matter what the query was. The dotted line shows the case where the retrievability scores are uniformly distributed—these data are simulated randomly here.

The Gini [12] coefficient gives a score from 0 to 1 indicating the degree of inequality. It is calculated as the area between the 45° unbroken line and the Lorenz curve divided by the total area underneath the 45° line. A score of 0 implies total equality; for total inequality, where only one document is ever retrieved, the coefficient approaches 1. In Fig. 7.1, the Gini coefficient for the unbroken line is 0. For the heavy, dashed line it is close to 1 and for lighter dashed line is 0.329. If we assume that the documents have been placed in order of non-decreasing retrievability according to some estimated measure  $\hat{R}(d_i)$  and that there are  $N$  documents in the collection we can define more formally the Gini coefficient as

$$1 - \frac{2}{N-1} \left( N - \frac{\sum_{i=1}^N i \cdot \hat{R}(d_i)}{\sum_{i=1}^N \hat{R}(d_i)} \right)$$

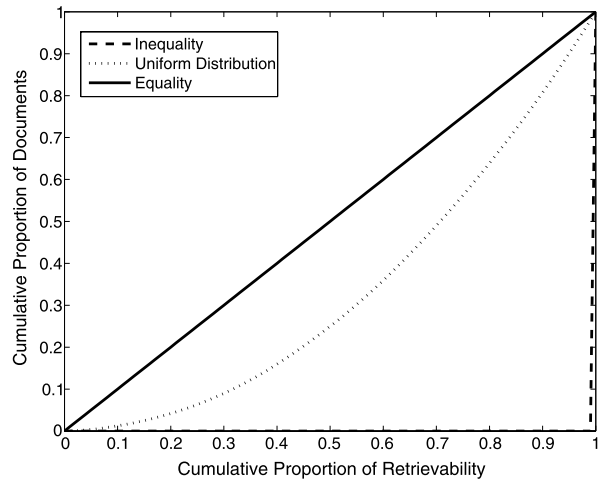
for discrete data, where the documents are indexed by non-decreasing order of  $\hat{R}$ .

### 7.3.4 The Retrievability Experiment

Having described the retrievability and access measures, we now summarize the steps undertaken to perform the retrievability analysis, which follows the methodology described in [5].

1. For each corpus of patent documents, we indexed the collection and removed stop words.
2. A large number of queries were automatically generated from the index (see the next section for the method of query generation).

**Fig. 7.1** An example of a Lorenz curve



3. For each IR model, each query was used to retrieve documents from the corpus. In Sect. 7.6 we describe the best-match, exact-match and hybrid models used in this study. The results from all queries were stored.
4. For each set of results for each retrieval model, the retrievability measurements were calculated for each document.
5. Given the retrievability scores for all documents the overall access measure was calculated (i.e. Gini coefficient) for each model. Here, we attribute a model with a lower Gini coefficient as providing better access to the collection.

The retrieval experiments and query generation was performed using the Lemur Toolkit 4.10 [21].

## 7.4 Query Generation

To obtain a reasonably accurate estimate of the retrievability scores a sufficiently large number of queries is required. This means that manual generation of queries is not practical and instead some algorithm is required. In this paper, we adopt the approach previously taken in [5], where a series of one-word and two-word queries were automatically generated given the documents in the collection. However, here, only two-word queries were chosen since one-word queries cannot show the effect of Boolean operators. Although the queries used in patent search are often longer than two words, we choose two-word queries because this means there will be just one Boolean operator and it thus affords a comparison between the use of AND and OR within the various IR models. Also, using only two-word queries makes the computation and estimation of the retrievability scores tractable.

Any set of generated queries must relate to the corpus under analysis (i.e. the queries need to contain terms that are in the documents). It is for this reason that query generation method extracts queries using the corpus index. The *collocation*

**Table 7.1** Examples of queries automatically generated by the collocation method

---

diaminopy rimidin
indexobjekte datenbestandes
leandro chemdb
dimethanesulphonate myleran
laktat malat
moxon schweda
menziesii mirb
finegold angelopoulou
oxalkylen mischpolymerisate
phenanthrenequinone phenylpropanedione

---

method used in previous studies [5] identifies common phrases in the text, and assumes that they are likely candidates for queries. A collocation is a pair of words which occur together more often than would be expected by chance. Queries were derived from collocations found in the document collection since they would tend to relate to some concept which simulates some user information need. The procedure for generating the queries has four distinct stages.

1. The set of all bigrams (pairs of consecutive words) are extracted from the index of the collection.
2. Those bigrams that occur less than a specified number of times are removed.
3. The adjusted point-wise mutual information measure (APWMI) [16] is calculated for each bigram and this is used to sort them in descending order.
4. The first 200 000 of this list of bigrams are disregarded since they contained a lot of stock phrases that were independent of the technical/topical content (e.g. *diagram shows*). The next  $n$  queries were then used for the experiment.

These steps are now explained in more detail.

For the purpose of indexing, only those parts of the patent dealing with technical details were used. Thus the sections which listed names and addresses as well as the copyright notice were omitted since these would give rise to common collocations which would not represent plausible queries. Stopwords and punctuation were also removed, as were terms that contained digits or had fewer than two letters. The text was case lowered but stemming was not applied. For each document, each consecutive pair of remaining words was considered as a bigram. Then, each distinct bigram was recorded and the number of occurrences counted. Only those bigrams which occurred at least 10 times were then considered. It is worth noting here that since the IR models under investigation are assumed to have the same indexer with only the matching function varying, a common set of queries was used for all models. Table 7.1 gives an example of 10 consecutive queries, which have been automatically generated.

### 7.4.1 *How Many Queries?*

A question arises as to how many queries should be used for the estimation of retrievability. Too few queries will give an inaccurate result, and in particular will mean that the estimated Gini coefficient will be much higher than the true estimate. Thus, we need to ensure that we use enough queries to obtain a reasonably accurate estimate of the Gini coefficient, even though it will not be exact. And while the estimate will not completely accurate it is asymptotically consistent, meaning that inaccuracy of the measurement will tend to decrease as the number of queries increases. Of course, there is a practical limit on the number of queries that can be used since each query requires a retrieval operation. Here we have used up to 1 million queries depending on the collection. In order to facilitate the comparison of models across different corpora, it is important that the document to query ratio (DQR) is held constant so that the estimates of the Gini coefficients will not be out of proportion, i.e. if more queries are used on one corpus than another, then its Gini coefficients will be closer to the exact values, which means that it is not possible to compare across corpora. For comparisons to be meaningful, the DQR should be kept constant so that the number of queries used is kept proportional to the number of documents in each corpus. Essentially, if corpus A is twice as large as corpus B, then it should have twice the number of queries.

## 7.5 Features for Improving Access

In previous work [7], we conducted a pilot study to determine the differences in retrievability over a number of popular IR models (best-match and exact-match models), and to ascertain to what extent documents were un-retrievable given these models. It soon emerged that there were marked differences in the retrievability of documents between the best-match and exact-match models. This led to the hypothesis that there were certain features of the best-match models, not shared with the exact-match models that increased access to the corpus by making the retrievability of documents more equal.

Three features of the best-match models were identified as potentially affecting retrievability. It should be noted that these features are also likely to improve effectiveness, but such an investigation is beyond the scope of this chapter. However, it was with effectiveness in mind that Fang et al. [11] identified a set of constraints that an IR model, ought to adhere to. The chosen features closely relate to some of these constraints. The three features identified were:

1. Sensitivity to Term Frequency,
2. Length Normalization,
3. Convexity.

Each is now considered in turn.

Whereas the traditional Boolean model considers only if a term is present or absent, best-match models such as BM25 and Tf-Idf take into account the number

of occurrences. A higher frequency of a given query term will make the document more relevant. This is codified in the Term Frequency Constraint (TFC1) in [11]. It is also common to take account of how frequently the term appears in the entire collection so that very commonly used vocabulary score less than the rarer words. This is consistent with their Term Discrimination Constraint (TDC) [11].

One consequence of incorporating term frequency into a model is that it will tend to score longer documents higher than shorter documents. Although we can argue that, all other things being equal, a longer document is more likely to be relevant since it will contain more information, there is a tendency to over-score longer documents. Thus many models incorporate some length normalization so that shorter documents are not unduly penalized. This is consistent with the Length Normalization Constraint (LNC1) [11].

Fang et al. [11] also attempt to define a ‘desirable characteristic’ of a retrieval model whereby the combination of two distinct query terms in the document will score higher than just occurrences of one or the other. We formalize this concept as *convexity* and provide the following definition. Let  $q$  be a query with two terms  $w_1, w_2$ . Let  $d_1$  and  $d_2$  be documents which yield precisely the same ranking score for the query  $q$ . Assume that  $d_1$  contains two occurrences of  $w_1$  and none of  $w_2$ . Assume also that  $d_2$  is identical to  $d_1$  except that the two occurrences of  $w_1$  are replaced with  $w_2$ . Now, let us assume a third document  $d_3$  which is identical to  $d_1$  except that only one occurrence of  $w_1$  is replaced with  $w_2$  so that it now contains one of each query term. An IR model will have convexity if, and only if, it always ranks  $d_3$  (strictly) higher than  $d_1$  and  $d_2$ .

One key feature of the traditional Boolean model is that it provides a crisp cut-off since each document will be assigned a score of 0 or 1. Adding sensitivity to term frequency means that the scoring of any document *cannot* have just two values. Nevertheless, a crisp cut-off would be possible if there were some attainable minimum score, say 0, representing the Boolean condition being false, and a positive value to represent that it is true.

## 7.6 Retrieval Models

For the purposes of our empirical study, we considered an exact-match retrieval model (i.e. the traditional Boolean model), two well established best-match models (i.e. Tf-Idf and BM25) and a number of hybrid models, which can accept a Boolean query but take into account some or all of the features introduced above. In an initial study [7], we considered several other hybrid models, but here we perform this larger study on a subset of the best performing retrieval models (i.e. those models that improved access). An IR system consists of an index and a matching model. Since we use the same index throughout the empirical study, the only variation between systems is the matching model.

### 7.6.1 Traditional Boolean Models

Given that the simulated queries consist of two terms, both of which are considered desirable in any retrieved document, they may only be combined in two ways, namely with an AND and an OR. As a true exact-match model there is no ordering imposed on the retrieved documents, so two possible methods of sorting the retrieved documents were chosen:

1. Chronological order—earliest document first,
2. Reverse Chronological order—latest document first.

We note here that Boolean AND will possess convexity since it clearly gives a higher ranking when both terms are present.

### 7.6.2 Tf-Idf

This is really a family of retrieval models where the acronym stands for Term Frequency–Inverse Document Frequency [19]. We shall use the notation  $c(w, d)$  to represent the counting function which yields the number of occurrences of word  $w$  in document  $d$ . The document frequency  $df(w)$  is the number of documents in the collection which contain at least one occurrence of  $w$ . There are several formulations for inverse document frequency (Idf). The one chosen here prevents the Idf from ever becoming 0, even if a term is present in every document. We have

$$Idf(w) = \log\left(\frac{N + 1}{df(w)}\right), \quad (7.3)$$

where  $N$  is the number of documents in the collection and, in the following,  $q$  is the query. The ranking function multiplies the occurrence of each query term in the document by the Idf measure:

$$\sum_{w \in q \cap d} idf(w) \cdot c(w, d). \quad (7.4)$$

This we will refer to as *standard* Tf-Idf and note that it is sensitive to term frequency but possesses neither the convexity property nor length normalization. It is, however, possible to address length bias by using pivoted normalization and define *normalized* Tf-Idf as

$$\sum_{w \in q \cap d} \frac{Idf(w) \cdot c(w, d)}{(1 - b) + b \cdot \frac{|d|}{avdl}}, \quad (7.5)$$

where  $|d|$  is the size of the document and  $avdl$  is the average document length. We set the parameter  $b$  to 0.75 to be the same as the BM25 model below.

If we consider retrieving all the documents whose matching score is greater than 0, we can see that this has the same effect as the Boolean OR since if any of the terms are present the ranking function will have a positive value. However,

the retrieved documents will nevertheless be sorted according to the frequencies of the query terms. If more query terms are present or these terms are rarer in the collection, it will score a document higher.

### 7.6.3 *Okapi BM25*

This is also a family of matching models and is often referred to as either Okapi or BM25. The original formulation [20] is based on a probabilistic model. However, various ad hoc changes have been advocated such as that proposed by Fang et al. [11]. It is also common to ignore the query factor of the original formulation (this can be achieved by allowing one of the parameters to approach infinity). The formulation used is:

$$\sum_{w \in q \cap d} \text{idf}(w) \cdot \frac{(k_1) \cdot c(w, d)}{k_1 \cdot ((1 - b) + b \cdot \frac{|d|}{\text{avdl}}) + c(w, d)}. \quad (7.6)$$

Two parameters in the model are set to the following value  $b = 0.75$  and  $k_1 = 1.2$  as is standard practice. As with Tf-Idf, BM25 has the intrinsic OR property and also takes account of term frequency. Because each additional occurrence of the same term makes a diminishing contribution to the overall score, this function also exhibits convexity. This model has been shown to perform well in TREC evaluations and, in particular, to outperform Tf-Idf.

### 7.6.4 *Filtered Term-Frequency Models*

The idea of constructing a best-match model which accepts a Boolean query is not new. Salton et al. [18] attempted this some years ago. However, their proposed model does not give a crisp cut-off, which we argued for in Sect. 7.2, and thus we seek other solutions. One reason we have proposed for using a Boolean query is that the use of the AND operator can actually prevent some documents being retrieved. We propose one approach to add conjunctivity to the two term-frequency models described above. A second approach, the harmonic model [7] was found to yield almost identical results, and so it is not included here. Thus we use a Boolean expression to filter the results of best-match function. That is, where the Boolean expression is true, the document is scored according to a best-match model such as Tf-Idf and BM25. Where the condition is false, the document is scored at 0. It is worth noting here that both BM25 and Tf-Idf will yield 0 if and only if no query terms appear in the document. This is not true for all best-match models and so this approach is only applicable to certain best-match models.

We assume that there is some Boolean query consisting of a set of query terms with the operators AND and OR. The words are extracted and used to calculate a term-frequency model except that if the Boolean expression is false, the matching

**Table 7.2** Models used in experiment

	No Convexity	Convexity
Term Presence	Chronological OR Reverse Chronological OR	Chronological AND Reverse Chronological Boolean Filter
Term Frequency	Standard Tf-Idf	Standard Tf-Idf with Boolean Filter
Term Frequency with Length	Normalized Tf-Idf	BM25 BM25 with Boolean Filter
Normalization		Normalized Tf-Idf with Boolean Filter

value is set to 0. If the Boolean expression contains only the OR operator, the matching value will be 0 when the Boolean expression is false anyhow. The filter only cuts when there are AND operators. This approach has been used before by Arampatzis et al. [1] on legal queries. The approach can be generalized to Boolean expressions containing the NOT. Terms that are required to be absent are kept within the Boolean expression and if they were present it would set the matching value to 0. However, such terms would not form part of the list of terms used to calculate the Tf-Idf or BM25 function. There is one theoretical drawback with the filtering approach, which is that the filter introduces a discontinuity in the matching function.

We note here that when used with an AND operator, the filtered models will exhibit convexity, whereas the OR operator will not, except for BM25, where convexity is present in both cases.

### 7.6.5 Summary of Models

Since the generated queries used have two terms each, we can consider there to be either an AND or OR operator between them. We note as stated above that applying a Boolean filter makes no difference for an OR operator. Only when AND is used is the result changed.

Table 7.2 summarizes the ten models used according to the presence or absence of the three features. Note that the models that are insensitive to term frequency are all exact match. We expect retrievability to improve towards the bottom of the table.

## 7.7 Results

We calculated the Gini coefficients of retrievability for each IR model given each corpora from the four patent offices (European, US, Japanese and World) which together make up the MAREC collection [17]. This allowed both comparison of the models, revealing the ones which can give greater access and also comparison between corpora. For this latter purpose, the number of generated queries has been chosen to keep the document to query ratio constant.



**Table 7.3** Summary statistics for MAREC collection and generated queries

Corpus	Number of Documents	Number of Queries	Document to Query ratio (DQR)
European (EP)	3,508,686	415,054	8.454
Japanese (JP)	8,453,560	1,000,000	8.454
United States (US)	5,639,471	667,112	8.454
World (WO)	1,784,980	211,151	8.454

Corpus	Number of Terms	Number of Unique Terms	Mean Document Length
European (EP)	4,936,283,816	11,412,080	1,406.88
Japanese (JP)	735,047,852	802,875	86.95
United States (US)	16,107,226,982	11,281,670	2,856.16
World (WO)	4,288,769,761	15,767,937	2,402.70

Summary statistics relating to the patents contained in the corpora are shown in Table 7.3 along with the number of queries used. The European and World patent documents were in English, French and German, with many documents containing more than one language. Other corpora were in English only. It should be noted that whereas some documents were the full version of the patent, others, specifically in the Japanese corpus, were short summaries or abstracts. The patent documents contained in the four corpora were semi-structured (in XML format) meaning that it was possible to extract automatically certain sections for indexing and not others.

The index used for retrieval was the same as the one used to generate queries; that is it contained only the technical sections of each patent document. Although it would have been more realistic to have used the entire document, an initial study performed on the World collection showed that the results were very similar. Thus to save the computing time and storage space of creating separate indices, a single index was used for each corpus. Three languages were used in the whole MAREC collection: English, French and German. Some documents had a mixture of languages. Thus a combined three-language stop word list was used. Stemming was not applied. We now address comparison both of the models and the corpora.

### 7.7.1 Comparison of Models

Table 7.4 gives Gini coefficients for the four corpora for all ten models and five retrievability measures for different values of  $\tau$ . For the first two corpora (EP, JP) the results are very similar to those in the pilot study [7].

1. The term-frequency sensitive models outperform corresponding models that capture only term presence. Specifically, standard Tf-Idf shows greater retrievability

**Table 7.4** Gini coefficients for all scoring functions

Model	OR Relation					AND Relation				
	10	20	50	100	200	10	20	50	100	200
European (EP)										
BM25	0.815	0.778	0.742	0.722	0.705	0.821	0.788	0.762	0.751	0.746
Tf-Idf Std.	0.965	0.947	0.916	0.889	0.861	0.916	0.892	0.861	0.842	0.828
Tf-Idf Norm.	0.946	0.919	0.876	0.840	0.806	0.876	0.846	0.817	0.802	0.792
Boolean Ch.	0.998	0.996	0.990	0.984	0.974	0.973	0.956	0.924	0.898	0.873
Boolean Rev.	0.998	0.996	0.991	0.984	0.975	0.974	0.957	0.925	0.899	0.874
Japanese (JP)										
BM25	0.712	0.609	0.487	0.423	0.384	0.709	0.601	0.462	0.374	0.305
Tf-Idf Std.	0.990	0.983	0.966	0.945	0.912	0.817	0.725	0.586	0.488	0.408
Tf-Idf Norm.	0.988	0.981	0.964	0.941	0.906	0.805	0.705	0.556	0.452	0.368
Boolean Ch.	1.000	1.000	0.999	0.998	0.996	0.963	0.933	0.863	0.788	0.696
Boolean Rev.	1.000	1.000	0.999	0.998	0.996	0.962	0.931	0.863	0.789	0.699
United States (US)										
BM25	0.873	0.842	0.804	0.779	0.755	0.886	0.864	0.840	0.826	0.814
Tf-Idf Std.	0.968	0.952	0.923	0.896	0.867	0.936	0.919	0.897	0.882	0.867
Tf-Idf Norm.	0.955	0.932	0.891	0.854	0.817	0.910	0.891	0.869	0.855	0.841
Boolean Ch.	0.996	0.993	0.986	0.979	0.969	0.970	0.953	0.927	0.907	0.886
Boolean Rev.	0.996	0.994	0.988	0.981	0.972	0.973	0.957	0.933	0.913	0.893
World (WO)										
BM25	0.846	0.812	0.770	0.746	0.734	0.890	0.888	0.892	0.897	0.901
Tf-Idf Std.	0.911	0.880	0.843	0.821	0.806	0.909	0.899	0.897	0.899	0.902
Tf-Idf Norm.	0.886	0.849	0.805	0.780	0.764	0.897	0.892	0.893	0.897	0.901
Boolean Ch.	0.979	0.966	0.938	0.905	0.865	0.909	0.898	0.896	0.898	0.902
Boolean Rev.	0.989	0.981	0.958	0.928	0.886	0.920	0.905	0.899	0.900	0.902

than Boolean OR. Also Tf-Idf with an AND filter outperforms the Boolean AND model.

2. Models with length normalization (BM25 and Normalized Tf-Idf) outperform standard Tf-Idf which is not normalized.
3. Models with convexity outperform the corresponding models without. In particular, standard BM25 outperforms standard Tf-Idf which implies that convexity is important when the OR operator is used; it is always present when we use the AND operator.

Thus when each of the model features is present, it gives a more equitable retrievability. For the US corpus, points 1 and 2 are also demonstrated but the story for point 3 is more ambiguous. For smaller numbers of documents retrieved

**Table 7.5** Mean occurrence of bigrams used in query set

Corpus	Mean Occurrence
European (EP)	250.23
Japanese (JP)	117.19
United States US	185.99
World (WO)	21.96

( $\tau = 10, 20, 50$ ) the AND operator gives better retrievability than OR, except for BM25 where the OR version has convexity anyway. However, as the number of documents retrieved increases 100 or 200, we observe that for the two Tf-Idf variants sometimes the OR outperforms AND.

For the WO corpus, points 1, 2 and 3 are true for the OR operator only. For the AND operator, the results are curious for two reasons. Firstly, the Gini measure does not increase as  $\tau$  increases; indeed it sometimes falls. Secondly, the Gini measures appear very similar for all models, which is a pattern not shown elsewhere. A further investigation shows that this can be attributed to the queries being somewhat different. Table 7.5 shows the mean occurrences of each bigram corresponding to a query in each entire corpus. For the WO corpus the bigrams are far less frequent and one therefore would expect them to be found in far fewer documents. Of course, each of the terms in any bigram could appear separately in any document but the fact these bigrams were selected by having a high APWMI measure implies that occurrences of the terms separately will also be rare. Thus for many of the queries in the WO corpus, there will be few documents which contain both terms and so few documents will be retrieved for each of these queries. This is confirmed in Table 7.6, which shows the mean number of times each document is retrieved. Note that this statistic is the same for each of the five matching functions. This raises the question as to whether the WO corpus has radically different properties from the other corpora or whether the collocation method has failed to find plausible queries. However, we leave this for future work.

Table 7.7 shows the frequency with which documents are retrieved. We note that where the Gini coefficient is very high, it corresponds to a large number of documents never being retrieved. One explanation of lack of retrievability consistent with this observation is that some documents are regularly pushed down the ranking

**Table 7.6** Mean frequency of retrieval for all scoring functions

Corpus	OR Relation					AND Relation				
	10	20	50	100	200	10	20	50	100	200
Number of Top Documents Retrieved ( $\tau$ )										
European (EP)	1.181	2.357	5.863	11.640	23.011	1.140	2.224	5.238	9.580	16.789
Japanese (JP)	1.183	2.366	5.913	11.823	23.634	1.179	2.337	5.648	10.745	19.835
United States (US)	1.171	2.319	5.663	11.078	21.589	1.108	2.094	4.563	7.783	12.692
World (WO)	1.178	2.349	5.830	11.501	22.184	0.914	1.447	2.048	2.367	2.568

**Table 7.7** Percentage of documents retrieved  $n$  times for each corpora using  $\tau = 50$ 

Model	OR Relation					AND Relation				
	How Often Retrieved									
	0	1	2-9	10-99	$\geq 100$	0	1	2-9	10-99	$\geq 100$
European (EP)										
BM25	36.68	13.67	31.13	18.33	0.18	40.45	13.75	29.73	15.88	0.20
Tf-Idf Std.	67.43	7.28	14.40	9.91	0.97	54.88	10.53	22.66	11.29	0.64
Tf-Idf Norm.	61.80	7.57	16.61	13.25	0.77	49.96	10.64	25.10	13.89	0.41
Boolean Ch.	85.53	6.03	5.84	2.08	0.52	62.54	12.89	16.65	7.01	0.91
Boolean Rev.	85.65	5.97	5.85	2.04	0.50	62.88	12.75	16.49	6.97	0.91
Japanese (JP)										
BM25	10.78	10.79	58.19	20.23	0.00	9.71	10.69	60.89	18.71	0.00
Tf-Idf Std.	78.66	6.77	8.97	4.12	1.47	19.30	13.39	47.34	19.98	0.00
Tf-Idf Norm.	77.22	7.34	9.67	4.28	1.49	16.12	13.21	50.82	19.86	0.00
Boolean Ch.	96.93	1.17	1.13	0.52	0.24	52.18	17.23	22.39	7.64	0.56
Boolean Rev.	96.93	1.16	1.13	0.53	0.25	46.86	17.20	25.25	9.66	1.02
United States (US)										
BM25	41.66	15.05	28.83	14.07	0.39	48.27	16.30	24.55	10.50	0.38
Tf-Idf Std.	68.38	7.27	14.23	9.24	0.89	58.26	13.68	19.69	7.71	0.64
Tf-Idf Norm.	65.61	7.94	16.23	9.74	0.47	53.68	14.64	21.93	9.21	0.54
Boolean Ch.	83.08	6.17	7.52	2.69	0.54	65.39	11.59	15.81	6.49	0.71
Boolean Rev.	84.05	5.94	6.97	2.52	0.51	66.59	11.44	15.13	6.13	0.71
World (WO)										
BM25	41.43	12.92	27.46	17.97	0.22	68.66	10.74	15.86	4.59	0.14
Tf-Idf Std.	54.02	9.52	21.30	14.56	0.60	69.32	10.62	15.42	4.47	0.16
Tf-Idf Norm.	48.70	10.55	23.52	16.89	0.34	68.87	10.69	15.74	4.56	0.15
Boolean Ch.	71.42	7.46	12.29	7.59	1.24	69.21	10.62	15.50	4.51	0.16
Boolean Rev.	74.55	7.83	10.94	5.54	1.13	69.47	10.75	15.27	4.34	0.17

by others, which also match the query. This would explain the very high numbers of non-retrieved documents when using the Boolean model where earlier or later documents always come before them and are thus more frequently retrieved.

### 7.7.2 Comparison of Corpora

Considering Table 7.4 again, we can see a marked difference between the corpora in terms of access. Not only do the Gini measures show different values for the same model, but also some models perform better on some collections than others. The

JP collection has potentially the greatest access if models with convexity are used, yet the worst level of access when using the Boolean OR. It should be noted that the JP collection has much smaller documents than the other three, since it consists mainly of short patent summaries, whereas the other corpora contain a mixture of summaries and full patents.

## 7.8 Conclusion

From the retrievability analysis that we have conducted, it appears that the hybrid models offer patent searchers the best of both worlds. On the one hand they accept a Boolean query and provide a crisp cut-off, thus fulfilling the requirements explained in Sect. 7.2. On the other hand, they provide improved access to the documents within the collection over the traditional Boolean model. This is because the hybrid models included the three features: term-frequency sensitivity, length normalization and convexity, which have been shown to improve access across the collection. Our analysis also shows that different models provide greater access depending on the collection, and so the choice of model is dependent on the corpora. This research provides an interesting starting point for the analysis and profiling of collections, such as patent corpora, and determining how easily the documents can be found given a particular system. In the case of patent searching, it is very important that the tools that patent searchers employ enable them to access all parts of the collection as easily as possible. If parts of the collection are not easily accessible, then this could lead to missing relevant documents, and doing so could be quite costly.

The idea of measuring access to a corpus is a relatively new one, when compared to precision and recall which have been around since the 1960s. Our measure of access, retrievability is just one of many such measures that could be derived to assess this attribute. We have compared a selection of IR models and identified those features of the IR system that can improve access. But this is by no means an exhaustive evaluation. What we have presented here is a feasible and well-defined framework for measuring access and it is for others in the patent domain and elsewhere to take these ideas further.

**Acknowledgements** This work described in this chapter was supported and partly funded by Matrixware. I would like to thank the Information Retrieval Facility for their computation services. I would also like to thank Leif Azzopardi, Tamara Polajnar, Richard Glassey and Desmond Elliott for their helpful comments and suggestions on how to improve this work.

## References

1. Arampatzis A, Kamps J, Koolen M, Nussbaum N (2007) Access to legal documents: Exact match, best match and combinations. In: TREC 2007: NIST special publication 500-274: The sixteenth text retrieval conference proceedings. NIST, Gaithersburg

2. Azzopardi L, Bache R (2010) On the relationship between effectiveness and accessibility. In: 33rd international ACM SIGIR conference on research and development in information retrieval, 19–23 Jul 2010, Geneva, Switzerland
3. Azzopardi L, Vinay V (2008) Accessibility in information retrieval. In: Advances in information retrieval ECIR 2008, Glasgow, UK, March 30–April 3. Springer, Berlin, pp 482–489
4. Azzopardi L, Vinay V (2008) Document accessibility: Evaluating the access afforded to a document by the retrieval system. In: Evaluation workshop at the European conference in information retrieval, Glasgow, UK, March 30–April 3
5. Azzopardi L, Vinay V (2008) Evaluation methods for information access tasks. In: CIKM 2008 proceedings of the 17th ACM international conference on information and knowledge management, California, US, 26–30 October. ACM Press, New York
6. Azzopardi L, Vanderbauwhede W, Joho H (2010) A survey of patent analysts' search requirements. In: Proceedings of the 33th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR 2010), Geneva, Switzerland, pp. 775–776
7. Bache R, Azzopardi L (2010) Identifying retrievability-improving model features to enhance boolean search for patent retrieval. In: Proceedings of the 1st international workshop on the advances in patent information retrieval
8. Bashir S, Rauber A (2009) Improving retrievability of patents with cluster-based pseudo-relevance feedback documents selection. In: Proceedings of the 18th ACM conference on information and knowledge management (CIKM2009), Hong Kong, November 2009. ACM, New York
9. Bashir S, Rauber A (2010) Improving retrievability of patents in prior-art search. In: Advances in information retrieval. Lecture notes in computer science, vol 5993, pp. 457–470
10. Bonino D, Ciaramella A, Corno F (2010) Review of the state-of-the-art in patent information and forthcoming evolutions in intelligent patent informatics. *World Pat Inf* 32(1):30–38
11. Fang H, Tao T, Zhai C (2004) A formal study of information retrieval heuristics. In: SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval. ACM, New York, pp 49–56
12. Gastwirth J (1972) The estimation of the Lorenz curve and Gini index. *Rev Econ Stat* 54:306–316
13. Hunt D, Nguyen L, Rodgers M (2007) Patent searching: Tools and techniques. Wiley, New York
14. Joho H, Azzopardi L, Vanderbauwhede W (2010). A survey of patent users: An analysis of tasks, behavior, search functionality and system requirements. In: Proceedings of the 3rd symposium on information interaction in context (IiX 2010) 54(3):306–316
15. Ma H, Chandrasekar R, Quirk C, Gupta A (2009) Improving search engines using human computation games. In: CIKM '09: Proceeding of the 18th ACM conference on information and knowledge management, pp 275–284
16. Manning C, Schütze H (1999) Foundations of statistical natural language processing. MIT Press, Cambridge
17. Matrixware research collection. <http://www.ir-facility.org/research/data/matrixware-research-collection>, Last visited 2010
18. Salton G, Fox E, Wu H (1983) Extended boolean information retrieval. *Commun ACM*, 1022–1036
19. Spärk Jones K (2004) A statistical interpretation of term specificity and its application in retrieval. *J Doc* 60(5):779–840
20. Spärk Jones K, Walker S, Robertson SE (2000) A probabilistic model of information retrieval: Development and comparative experiments (parts 1 and 2). *Inf Process Manag* 36(6):493–502
21. The lemur toolkit. <http://trec.nist.gov/data.html>, Last visited 2010
22. Tseng YH, Wu YJ (2008) A study of search tactics for patentability search: A case study on patent engineers. In: PaIR '08: Proceeding of the 1st ACM workshop on patent information retrieval. ACM, New York, pp 33–36

# Chapter 8

## Measuring Effectiveness in the TREC Legal Track

Stephen Tomlinson and Bruce Hedin

**Abstract** In this chapter, we report our experiences from attempting to measure the effectiveness of large e-Discovery result sets in the TREC Legal Track campaigns of 2007–2009. For effectiveness measures, we have focused on recall, precision and  $F_1$ . We state the estimators that we have used for these measures, and we outline both the rank-based and set-based approaches to sampling that we have taken. We share our experiences with the sampling error in the resulting estimates for the absolute performance on individual topics, relative performance on individual topics, mean performance across topics, and relative performance across topics. Finally, we discuss our experiences with assessor error, which we have found has often had a larger impact than sampling error.

### 8.1 Introduction

In this chapter, we report our experiences with measuring the effectiveness of approaches to electronic discovery search in the legal domain, which has many challenges in common with evaluating patent search. In particular, high recall is demanded in both of these domains, but the relevant documents may be a tiny fraction of the collection, making it difficult for sampling-based approaches to estimate the measures accurately.

Our experience comes from our involvement with the Legal Track of the Text Retrieval Conference (TREC), which started in 2006 [3], with the goal of creating standard tests for electronic discovery (e-Discovery) requests. Recall is a primary concern in e-Discovery, as there is a legal obligation to return, to an extent commensurate with a reasonable good-faith effort, *all* evidence relevant to the request. Precision is also important, however, in order to reduce cost and prevent the unnecc-

---

S. Tomlinson (✉)  
Open Text Corporation, Ottawa, Ontario, Canada  
e-mail: [stomlins@opentext.com](mailto:stomlins@opentext.com)

B. Hedin  
H5, 71 Stevenson St., San Francisco, CA 94105, USA  
e-mail: [bhedin@h5.com](mailto:bhedin@h5.com)

**Table 8.1** Overview of the referenced TREC Legal Track tasks

Task	$ D $	Topics	Type	max $ S $	Runs	Judged	$estRel(D)_{high}$
2007 Ad Hoc	6,910,192	43	rank	25,000	68	488–1000	77,467
2008 Ad Hoc	6,910,192	26	rank	100,000	64	493–900	658,399
2009 Batch	6,910,192	10	rank	1,500,000	10	1250–2500	1,046,833
2008 Interactive	6,910,192	3	set	6,910,192	5	2500–6500	786,862
2009 Interactive	569,034	7	set	569,034	4	2729–3975	26,839

essary release of information. Effective e-Discovery continues to be a challenging problem [2, 10].

The TREC Legal Track continued in 2007 [16], 2008 [11] and 2009 [8], each year running between one and three tasks. In this chapter, we refer to just five of these tasks, listed in Table 8.1, for which we were the lead coordinators for the task design, focusing particularly on the details of the sampling and measure estimation. Herein, we briefly summarize the details of these tasks that are necessary to understand the measurement approaches. More details on the TREC Legal Track are readily available in the online track overview papers [3, 8, 11, 16].

For each task, there was a document set  $D$  to search, either the IIT CDIP (Illinois Institute of Technology Complex Document Information Processing) collection [9], which consisted of 6,910,192 documents released by seven US tobacco companies, or the TREC 2009 Enron collection [8], which consisted of 569,034 e-mail messages (with attachments) from the mailboxes of approximately 150 employees of Enron Corporation. One can see which collection was used for each task based on the size of the collection ( $|D|$ ) listed in Table 8.1.

For each task, there was a set of test topics (approximately 50 new ones each year, though the final number was typically lower because some topics were not assessed in time for the track’s deadline that year). Each topic consisted of a multi-paragraph background complaint and a one-sentence request for documents to produce; for example, for topic #74, the (fictitious) complaint alleged infringement of a patent of a product for ventilating smoke, and the one-sentence request was “All scientific studies expressly referencing health effects tied to indoor air quality.” The number of test topics for a task ranged from three to 43; the final number for each task (excluding those discarded because of incomplete assessment) is in the “Topics” column of Table 8.1.

We refer to three of the five tasks (2007 Ad Hoc, 2008 Ad Hoc, 2009 Batch) as *rank-based* tasks, as per the “Type” column of Table 8.1. In these tasks, the test systems were typically automated systems, and they were required to specify a ranking of the documents for each topic based on the system’s opinion of their probability of relevance to the request. (In the “Ad Hoc” tasks, the topics were new ones that the systems were seeing for the first time, whereas in the “Batch” task, the topics were re-used from previous years and the systems could use past judgments of relevant and non-relevant documents to train batch filtering techniques.) Furthermore, for various bandwidth reasons, the rank-based tasks had a maximum submission depth



(e.g., 100,000 documents per topic in the 2008 Ad Hoc task) as listed in the “max  $|S|$ ” column of Table 8.1, which we hoped would provide sufficient coverage of the relevant documents.

We refer to the other two of the five tasks (2008 Interactive and 2009 Interactive) as *set-based* tasks (as per the “Type” column of Table 8.1). In these tasks, the test submissions were typically produced by an interactive (human-in-the-loop) process, and for each topic included just the documents that were considered relevant to the request, without specifying a ranking of the documents. In part because there were fewer test topics, there was no limit on the submission size (besides  $|D|$ , the size of the collection itself).

The remaining columns of Table 8.1 are as follows. The “Runs” column specifies the largest number of submissions received for any topic of the task; note that for the set-based (Interactive) tasks, participants were not required to submit results for every topic, unlike for the rank-based tasks. The “Judged” column specifies the smallest and largest number of documents judged for any topic of the task. And the “ $estRel(D)_{high}$ ” column specifies the largest estimated number of relevant documents for any topic of the task (based on the methodology discussed in Sect. 8.3 below).

We see that the number of relevant documents for a topic (sometimes more than 1 million) could far exceed the number of documents that we could judge for a topic (at most a few thousand). In the following sections, we describe the approaches we took to estimating the effectiveness measures and reflect upon how well the approaches met the various task goals. We also attempt to identify what evaluation challenges remain.

## 8.2 Effectiveness Measures

To gauge the effectiveness of a result set for a test topic, we focused on the well-known recall, precision and  $F_1$  measures [17].

If we had complete knowledge of which documents were relevant and non-relevant for a topic, we could calculate the recall, precision and  $F_1$  of a result set by using the following definitions:

- $D$             The set of documents in the collection.
- $S$              The subset of  $D$  whose effectiveness we wish to measure.
- $Rel(S)$      The set of relevant documents in  $S$ .
- $Non(S)$      The set of non-relevant documents in  $S$ .
- $Recall(S)$    The recall of  $S$ :

$$Recall(S) = \frac{|Rel(S)|}{|Rel(D)|}. \quad (8.1)$$

- $Prec(S)$     The precision of  $S$ :

$$Prec(S) = \frac{|Rel(S)|}{|Rel(S)| + |Non(S)|}. \quad (8.2)$$

$F_1(S)$  The  $F_1$  of  $S$ :

$$F_1(S) = \frac{2 * Prec(S) * Recall(S)}{Prec(S) + Recall(S)}. \quad (8.3)$$

Note:  $F_1(S)$  is 0 if either  $Prec(S)$  or  $Recall(S)$  is 0.

For ranked result sets, we can likewise gauge effectiveness at any particular cutoff depth  $K$  (remembering to pad the set with non-relevant documents if the set contained fewer than  $K$  documents in order to not overstate Precision@ $K$  or  $F_1@K$ ). Also, in our 2008 and 2009 rank-based tasks, we required our submissions to specify the depth  $K$  for each topic at which the system believed the  $F_1$  measure would be maximized, allowing both set-based and rank-based evaluation.

### 8.3 Estimators

In practice, we did not have the resources to judge all of the documents for each topic (almost 7 million documents for most of our tasks). The traditional TREC approach is simply to judge a pool of the top-ranked documents from various systems [7], but it was apparent from sampling experiments in 2006 [3, 13] that, for most of our topics, the number of relevant documents far exceeded the number that could be judged. Other TREC tracks had also been encountering issues with traditional TREC pooling [4].

In 2007, we started to use a deeper sampling approach to estimate the measures. It was based in part on the approach used to estimate “inferred average precision” (infAP) [22] in the TREC 2006 Terabyte Track [5]. Our main extension was to sample different parts of the collection with different probabilities (we defer our discussion of the sampling approaches to Sect. 8.4). Other researchers independently made a similar extension [1].

The estimators we have used for recall, precision and  $F_1$  are defined by

$d$	A document in $D$ .
$p(d)$	The inclusion probability of $d$ . i.e., the probability of selecting document $d$ for judging.
$JudgedRel(S)$	The set of documents in $S$ which were judged relevant.
$JudgedNon(S)$	The set of documents in $S$ which were judged non-relevant.
$estRel(S)$	The estimated number of relevant documents in $S$ :

$$estRel(S) = \sum_{d \in JudgedRel(S)} \frac{1}{p(d)}. \quad (8.4)$$

Note:  $estRel(S)$  is 0 if  $|JudgedRel(S)| = 0$ .

$estNon(S)$	The estimated number of non-relevant documents in $S$ :
-------------	---

$$estNon(S) = \sum_{d \in JudgedNon(S)} \frac{1}{p(d)}. \quad (8.5)$$

Note:  $estNon(S)$  is 0 if  $|JudgedNon(S)| = 0$ .

$estRecall(S)$  The estimated recall of  $S$ :

$$estRecall(S) = \frac{estRel(S)}{estRel(D)}. \quad (8.6)$$

$estPrec(S)$  The estimated precision of  $S$ :

$$estPrec(S) = \frac{estRel(S)}{estRel(S) + estNon(S)}. \quad (8.7)$$

Note:  $estPrec(S)$  is undefined if  $(estRel(S) + estNon(S)) = 0$ .

$estF_1(S)$  The estimated  $F_1$  of  $S$ :

$$estF_1(S) = \frac{2 * estPrec(S) * estRecall(S)}{estPrec(S) + estRecall(S)}. \quad (8.8)$$

Note:  $estF_1(S)$  is 0 if either  $estPrec(S)$  or  $estRecall(S)$  is 0.

The  $estRel(S)$  and  $estNon(S)$  formulas for estimating the number of relevant and non-relevant documents (respectively) use the Horvitz–Thompson estimator, which is unbiased [12]. (We do not claim, however, that our estimators for recall, precision and  $F_1$  are mathematically unbiased, because they involve ratios of estimators.)

We also have looked at alternative estimators that correct for obvious overestimates; for example, if  $estRel(S)$  is greater than  $|S| - |JudgedNon(S)|$ , then it must be an overestimate, and so reducing the estimate to  $|S| - |JudgedNon(S)|$  must reduce the error. We actually have used such alternative estimators in our rank-based tasks, and the formulas are stated in the 2007 track overview [16]. However, these alternative estimators bias the estimates low on average because only overestimates are improved; underestimates are left unchanged. In our experience, the alternative estimators have made little material difference, so we just present the simpler estimators in this chapter.

Another aspect to be accounted for that we have encountered in running our evaluations concerns what we have termed “gray” documents, which are documents that were drawn by sampling for assessment, but on which the assessor could not render a relevance judgment. This could occur for any of a number of reasons, such as a technical issue prevented a legible display of the document image, or the document was longer than 300 pages (which was more than we required an assessor to review for one document), or the document was in a language other than English. When reporting results, we have reported for each submission  $S$  an estimate of what percentage of  $S$  was gray documents; typically this percentage has been less than 2%, though we have seen as high as 13% from an approach that favored long documents [16]. The estimators we have given here for recall and precision essentially behave as if the gray documents had been omitted from both the full collection  $D$  and the result set  $S$ .

## 8.4 Sampling Approaches

This section describes how we sampled the collection, i.e., chose the  $p(d)$  values, in the various tasks.

As pointed out in related work on estimation approaches [1], the choice of  $p(d)$  does not affect the expected values of the estimators, but it can affect the variance and hence the accuracy of the estimates. Generally, we have chosen the  $p(d)$  values based on the submissions received (as described below) in hopes of minimizing the estimation error for the submissions. Future result sets can also be scored using the estimators, i.e., our test collections are reusable in principle, though the error bar for non-participating runs may be higher as the  $p(d)$  values may not be as suitable for them. We have not to date attempted a “system omission” study [23] for our test collections (i.e., a study in which we simulate how the estimated scores would have changed if one of the participating systems had not been included) which could be indicative of how reusable the collections may be in practice.

### 8.4.1 Rank-Based Sampling

As a concrete example of rank-based sampling, this section focuses on the 2008 Ad Hoc task.

As shown in Table 8.1, the rank-based 2008 Ad Hoc task had 26 test topics. Although the collection contained almost 7 million documents, for space and bandwidth reasons we only allowed participants to submit their top-ranked 100,000 documents for each topic, hoping that would be enough to include all of the relevant documents. The 10 participating groups submitted a total of 64 experimental runs.

For each topic, we created a pool  $P$  from all of the submitted documents. For each  $d \in P$ , we defined  $hiRank(d)$  to be the highest rank (where 1 is highest, 2 is 2nd highest, etc.) at which any of the 64 systems ranked the document. Then we set  $p(d)$  as follows:

$$\text{If } (hiRank(d) \leq 5) \quad \text{Then } p(d) = 1.0 \quad (8.9)$$

$$\text{Else } p(d) = \min \left( 1.0, \left( \left( \frac{5}{100000} \right) + \left( \frac{C}{hiRank(d)} \right) \right) \right) \quad (8.10)$$

The value  $C$  was chosen so that the sum of the  $p(d)$  values (for all  $d \in P$ ) was the number of documents that could be judged (typically 500 documents were judged for each topic).

This  $p(d)$  formula was intended to support (almost) equally accurate estimates regardless of the chosen depth  $K$ . One can see that at any depth  $K > 5$ , the smallest  $p(d)$  involved would be at least  $C/K$ , the same as if doing simple random sampling of at least  $C$  documents from the set of  $K$  documents. Unfortunately, for our 26 test topics, the  $C$  values turned out to range from just 1.7 to 4.4, which was lower than we had hoped. We discuss the implications for sampling error further in Sect. 8.5.

For documents  $d$  that were not in the pool,  $p(d)$  was 0. (Actually, we did draw a small random sample from the documents outside of the pool for separate analysis, which we discuss later in Sect. 8.6, but we did not use these for estimation because this sampling was deemed too coarse to be sufficiently accurate.) Hence our estimators actually were just estimating recall from the pool  $P$ . For estimating

recall, this approach essentially follows the traditional TREC approach of assuming all unpooled documents are non-relevant. For estimation of precision and  $F_1$ , however, for future result sets that might contain documents outside of the pool, our estimators behave not as if the unpooled documents were non-relevant, but as if the unpooled documents had been omitted from the result set.

### 8.4.2 Set-Based Sampling

As a concrete example of set-based sampling, this section focuses on the 2008 Interactive task.

As shown in Table 8.1, the set-based 2008 Interactive task had three test topics. Participants submitted just the set of documents that they considered relevant for a topic, without ranking the documents. There was no limit on the size of the submission set for a topic (other than the number of documents in the collection, 6,910,912). At most five submissions were received for any topic.

To assign the  $p(d)$  values, the full collection  $D$  was stratified. To use topic #103 as a concrete example, which received five submissions (one of which was actually a composite submission formed by pooling the results of 64 Ad Hoc submissions for the topic), 32 strata (from  $2^5$ ) were created as follows. The first stratum consisted of documents included in all five submissions (the “All-R” stratum, or “RRRRR”). The next stratum consisted of documents included in submissions 1–4 but not submission 5 (the “RRRRN” stratum). The next stratum consisted of documents included in submissions 1–3 and 5 but not submission 4 (the “RRRNR” stratum). And so on. The final stratum (stratum #32) included all of the documents that were not in any submission (the “All-N” stratum, or “NNNNN”).

Within a particular stratum  $S_s$ ,  $n_s$  documents were chosen to be judged (using simple random sampling without replacement). Typically  $n_s$  was chosen proportionally to  $|S_s|$  (the number of documents in the stratum), except that larger strata, particularly the “All-N” stratum, were sampled somewhat less densely than their full-population size would dictate, in order to ensure that we were able also to sample a sufficient number of documents from the smaller strata. For the purposes of the estimator formulas, for  $d \in S_s$ ,  $p(d)$  was  $n_s/|S_s|$ . As a concrete example, for topic #103, there were 6,500 judgments, and most strata had  $p(d)$  close to 0.008 (1 in 125), but the “All-N” stratum, which was 83% of the collection, was only assigned 25% of the samples ( $n_s = 1,625$ ), hence its documents’  $p(d)$  was just 0.00028 (approximately 1 in 3,500).

The 2008 Interactive task also introduced the practice of assigning multiple assessors to a topic. The documents to judge were allocated randomly to the available assessors; typically each assessor was responsible for a bin of 500 documents. If a bin was not completely assessed by the track’s assessment deadline, it was discarded, and the  $n_s$  and hence  $p(d)$  values of affected strata were reduced accordingly before the judgments were released.

**Table 8.2** Confidence intervals of some 2008 interactive task submissions

$ Judged(D) $	$ S $	$estRecall(S)$	$estPrec(S)$	$estF_1(S)$
6,500	608,807	0.624 (0.579, 0.668)	0.810 (0.795, 0.824)	0.705 (0.676, 0.734)
4,500	546,126	0.314 (0.266, 0.362)	0.328 (0.301, 0.355)	0.321 (0.293, 0.349)
2,500	689,548	0.345 (0.111, 0.580)	0.023 (0.014, 0.032)	0.043 (0.026, 0.060)

## 8.5 Sampling Error Analysis

In this section, we discuss our experiences with sampling error, i.e., the limitations on accuracy resulting from judging just a sample of a population instead of the full population. In Sects. 8.5.1 and 8.5.2, the sampling error arises from not having judged every document for a topic. In Sects. 8.5.3 and 8.5.4, the sampling error arises from having a limited number of test topics.

### 8.5.1 Absolute Performance on One Topic

For the stratified sampling approach described in Sect. 8.4.2, we have developed confidence interval formulas for  $estRecall(S)$ ,  $estPrec(S)$  and  $estF_1(S)$ ; these run several pages and are available in the 2008 track overview [11]. The formulas were developed in part by consulting textbook approaches [12]. Recent work suggests that these confidence intervals may be wider than they need to be and we are considering revisions to them [20]. However, we believe the formulas referred to here are still serviceable, if conservative, in their current state.

We should emphasize that all of the confidence intervals in this chapter are just accounting for the uncertainty arising from sampling error. We are not in this chapter attempting to construct confidence intervals that account for any other type of uncertainty, such as the uncertainty of whether the assessor was correct in his or her judgment of the relevance or non-relevance of each sampled document. (We investigate the impact of assessor errors separately in Sect. 8.6.)

Here, we just review some examples of the confidence intervals to give an idea of what widths were attained. Table 8.2 shows example confidence intervals for one submission for each topic of the 2008 Interactive task, in descending order by the number of judgments for the topic: 6,500 judgments for topic #103, 4,500 judgments for topic #102, and 2,500 judgments for topic #104. For example, the first row shows that, for a submission  $S$  of 608,807 documents, the estimated recall was 0.624, with 95% confidence interval of (0.579, 0.668).

Of course, examples cannot show the full picture, and it would be incorrect to suggest that the only factor in confidence interval size is the number of judgments. The overall yield of a topic (i.e.,  $|Rel(D)|/|D|$ ), for example, can also have a significant impact on the width of confidence intervals, with higher-yielding topics generally enabling narrower confidence intervals. Of all the strata, the one that poses the greatest sampling challenge is the All-N stratum (the stratum containing documents

no team identified as relevant). It is a challenge because the density of relevant material in this stratum is generally very low, making it hard to obtain, via sampling, precise estimates of the true density in the stratum; and this challenge generally becomes more acute as the overall yield of a topic gets lower. As a result, the lower the yield of a topic, the greater the sampling error contributed from the All-N stratum, and so the greater the width of the confidence intervals associated with our estimates of the full-population yield and of recall. (Note that, for the topics reported in Table 8.2, the estimated yield (i.e.,  $estRel(D)/|D|$ ) of the topic with 6,500 judgments was 0.114 of the full collection; the estimated yield for the topic with 4,500 judgments was 0.081; and the estimated yield for the topic with 2,500 judgments was 0.007.)

Nevertheless, from these examples, we can at least say that there are circumstances in which, with 4,500 judgments, we can obtain confidence intervals for recall less than 0.10 wide, and circumstances in which, with 2,500 judgments, we can obtain confidence intervals for recall that are more than 0.46 wide.

While we have not listed examples here from the same topic, the confidence intervals for the submissions were often narrow enough to not overlap the confidence intervals of any of the other submissions. For example, for the five submissions for topic #103, none of the confidence intervals for recall had any overlap of each other.

For the rank-based approach described in Sect. 8.4.1, for which typically there were just 500 judgments per topic, we have not to date computed confidence intervals for individual topic estimates, but it seems apparent from the low  $C$  values mentioned in Sect. 8.4.1 that the sampling error would be large in some cases. Large sampling error on individual topics does not imply that the test data are not useful, however, as discussed in the following sections.

### 8.5.2 *Relative Performance on One Topic*

While one of our goals was to provide reasonable estimates of the absolute values of the metrics, for comparing particular experimental approaches it can suffice to just estimate the difference in scores of the approaches. Sometimes the difference can be estimated much more accurately than the confidence intervals for the absolute values may suggest.

For example, suppose set  $S_1$  is estimated to have a recall of 0.62 with confidence interval (0.58, 0.66), and set  $S_2$  is estimated to have a recall of 0.64 with confidence interval (0.60, 0.68). One might conclude that the recall of  $S_1$  and  $S_2$  are not statistically distinguishable because their confidence intervals overlap. However, if one noticed that  $S_2$  was a superset of  $S_1$  and that there were relevant documents in  $S_2$  that were not in  $S_1$ , then one would know that the recall of  $S_2$  must be greater than that of  $S_1$  despite the overlap in the confidence intervals.

Strict subsets and supersets can arise in practice when comparing sets that result from Boolean queries. In particular, the Ad Hoc tasks of the TREC Legal Track included a reference Boolean negotiation for each test topic in which typically the

requesting party would argue for broadening the query and the responding party would argue for narrowing the query.

In general, one can also analyze differences of sets that overlap without one containing the other. (To date, however, we have not attempted to develop confidence interval formulas for such differences in scores, leaving this as future work.)

### ***8.5.3 Mean Performance Across Topics***

Sometimes there is interest in the average performance of an approach. For example, in the 2008 Ad Hoc task, for each of the 26 test topics, there was (as just mentioned) a reference Boolean negotiation, and the Boolean query initially proposed by the responding party was found to average just 4% recall, while the Boolean counterproposal by the requesting party was found to average 43% recall, and the resulting consensus query was found to average 33% recall. These average scores give us a feel for the typical negotiation in that it seems that the respondent's initial proposal was typically a very narrow query compared to the requester's rejoinder or resulting consensus.

One can compute approximate 95% confidence intervals for means by adding plus or minus twice the square root of the variance (assuming that there are at least 25 or so topics, and that the topics are independent). For example, the approximate confidence interval for the 33% average recall of the consensus negotiated query over the 26 test topics of the 2008 Ad Hoc task was (21%, 45%). The noisier the individual topic estimates, the higher the variance will tend to be, increasing the width of the confidence interval. Increasing the number of test topics will usually reduce the width of the interval.

Given a fixed assessment budget, there is a tradeoff between how many topics can be assessed and how many judgments can be made per topic. In our case, for the Interactive tasks, it would not have been practical to create a lot of topics because few participants would have time to perform an intensive interactive approach for all of them, so we focused on making the evaluation for the small number of topics as accurate as possible. For our Ad Hoc tasks of 2007 and 2008, in which the participating systems were typically automated, we followed the traditional TREC practice of creating enough topics to support averaging, albeit at the expense of accuracy on individual topics. In the 2009 Batch task, we reduced the number of topics, and while the primary reason was the bandwidth limitations of dealing with the increase in the allowed result set size, we also hoped that this tradeoff point would allow better failure analysis on individual topics (as discussed further in the next section). The "Million Query Track" at TREC [1] has explored the other extreme, creating more than a thousand test queries but judging only forty or so documents for each, in hopes of more accurate estimation of average performance.



### 8.5.4 *Relative Performance Across Topics*

Just as one can compute approximate confidence intervals for mean scores, one can compute approximate confidence intervals for the mean *difference* in score between two approaches. (The method given in the previous section, when applied to differences, is approximately the same as the popular paired *t*-test, which tends to be fairly accurate even if the differences are not normally distributed because of the Central Limit Theorem.) When zero is not in the confidence interval, the difference in the mean score is considered to be “statistically significant.”

For the 2007 Ad Hoc task (of 43 test topics), one study [14] compared 14 pairs of experimental approaches, thresholding relevance-ranked sets at depth  $B$  (the number of matches of the reference Boolean query); it found that three of the 14 differences in estimated recall, and seven of the 14 differences in estimated precision, were statistically significant. For the 2008 Ad Hoc task (of 26 test topics), a followup study [15] compared 15 pairs of experimental approaches; it found that three of the 15 differences in estimated recall@ $B$ , and three of the 15 differences in  $F_1$ , were statistically significant. These results indicate that the test collections for these tasks do sometimes support the discerning of statistically significant differences.

What is often more insightful than comparing mean scores across topics is to look at how often one approach substantially outscores another. Analyzing the largest differences can often lead to a better understanding of when one approach will outperform another. Such an investigation can also be interpreted as conducting “failure analysis” for the lower-scoring approach.

For example, in the 2007 Ad Hoc task, a study [14] compared the performance of the reference Boolean query to a relevance-ranked vector of the same keywords, thresholding the relevance-ranked retrieval set at the number of matches of the reference Boolean query. The Boolean query was found to have the higher estimated recall for 26 of the test topics, while the vector query scored higher on just 16 of the topics, and there was one tie. Why was the Boolean query often more successful? The largest difference was on topic #58, regarding “health problems caused by [high-phosphate fertilizers]”, for which the estimated recall of the Boolean query was 94% while for the vector query it was just 8%. Despite the potentially large sampling errors, it seemed clear from looking at some of the hundreds of judgments for the topic that the Boolean query was more successful for this topic because it required a term beginning with “phosphat” to be in the document, whereas the vector approach favored a lot of non-relevant documents that did not mention the key “phosphat” concept as it was only one of 21 terms in the vector form of the query. Finding good examples of when approaches differ may lead to a better understanding of when to use one approach or the other, or to the development of generally better approaches.

## 8.6 Assessor Error Analysis

In this section, we discuss our experiences with assessor error, which has proven to be a serious issue to address in order to accurately estimate recall, arguably even more important than sampling error.

The Interactive tasks of 2008 and 2009 included an adjudication phase in which the participants could appeal any judgment by the first-line assessor to the “Topic Authority” for the topic (whose judgment the initial assessor was attempting to replicate); the Topic Authority then rendered a final relevance judgment on all documents so appealed. For nine of the 10 test topics (all three from 2008, and six of seven from 2009), the estimated number of relevant documents ( $estRel(D)$ ) was lower after the adjudication phase, indicating that the initial assessors typically generated a lot of false positives. For example, for topic #103 in 2008,  $estRel(D)$  was 914,528 before adjudication and 786,862 after adjudication, a drop of 127,666 in a collection of 6,910,192 documents, suggesting that the false positive rate was approximately 2% of the collection.

We also have found evidence of a false positive rate in the Ad Hoc tasks of 2007 and 2008, even though they did not have an appeal process. As mentioned in Sect. 8.4.1, for these tasks we drew a random sample from the documents that no system submitted. Of these, we found that approximately 1% were judged relevant. When we personally reviewed some of these relevant judgments, almost all of them looked non-relevant to us (and we think the original assessors would agree with us in retrospect, though we regret that we did not reserve time with them to ask about particular judgments). This result suggests that there was a false positive rate of approximately 1% for unsubmitted documents.

A standout example of the impact of a small false positive rate was observed in topic #51 of the 2009 Batch task. For this task, there was a set of training judgments from a previous use of the topic, for which  $estRel(D)$  was 95. But with the new judgments for the topic in 2009,  $estRel(D)$  was 26,404. Most of the difference in these estimates came from just three relevant judgments in 2009, whose weights were approximately 8,000 each (from  $1/p(d)$ ) as no system retrieved them in their top-700,000 results. Our own review of these three documents suggests that they were false positives. We suspect that the original estimate of 95 was reasonably accurate, i.e., that relevant documents were just 0.001% of the collection for this topic. Hence a false positive rate of even 0.1% would lead to a huge overestimate of the number of relevant documents, and hence the recall of good result sets would be dramatically underestimated.

In the 2009 Interactive task, we found dramatic changes in the scoring of the result sets after the appeals. (The appeals typically corrected both false positives and false negatives.) In particular, for four of the seven topics, a participant’s (estimated)  $F_1$  score increased by more than 0.50 after the appeals; for example, on topic #201, the  $F_1$  of submission  $W$  increased from 0.07 to 0.84 after the appeals, and on topic #204, the  $F_1$  of submission  $H$  increased from 0.17 to 0.80 after the appeals.

Furthermore, the appeals in the 2009 Interactive task did not just change the absolute scores. For four of the seven topics, there were changes in the rankings of

the result sets (based on  $F_1$ ) after the appeals. One dramatic example of a re-ranking was on topic #205, for which the (estimated)  $F_1$  of submission  $C$  was higher than that of submission  $E$  before appeals (0.46 vs. 0.25), but lower after appeals (0.43 vs. 0.61), with both differences being statistically significant based on the lack of overlap in the confidence intervals.

While past studies have typically found only minor differences in system rankings from assessor differences [19], an exception has been noted in the past for manual runs involving relevance feedback [7, 18]. Of course, many of the Interactive task submissions were constructed with human assessing as part of the process, so our finding of the appeals affecting Interactive submission rankings appears to be consistent with past findings. (We note that a recent simulation-based study suggested more generally that false positives tend to cause larger differences in system rankings than false negatives [6].)

It seems clear from our experience, whether from the perspective of absolute scores or relative scores, that when a small number of judgments can substantially impact the scores, an evaluation needs to build into its process a way to deal with assessor error. Our experience with allowing the participants to appeal seems to have been reasonably successful for several of the test topics, but we have also seen topics with relatively few appeals which we suspect was not because those topics had a lower error rate but because appealing requires a lot of effort that not all participants are willing to undertake. In future evaluations, we are considering modifying the appeal process, perhaps to estimate the impact of appeals based on sampling, including automatically appealing a sample of all of the judgments, as suggested by a recent study [21].

## 8.7 Conclusion and Future Work

Our aim in this chapter was to summarize our approaches, look back upon how well the various task goals were achieved, and identify what challenges remain. In the set-based tasks, we found that our resulting confidence intervals for recall, precision and  $F_1$  were at least sometimes sufficient to distinguish differences between experimental approaches. In the rank-based tasks, we found that the estimation approaches were at least sometimes sufficient to identify statistically significant mean differences and conduct failure analysis. How to sample more efficiently and how best to quantify the estimated scores, including differences in scores, remain as challenges, particularly so in the case of low-yielding topics (which may be the typical circumstance in patent information retrieval). We also have found a lot of evidence that assessor error is an issue that cannot be ignored. How to best reduce these errors, or how to account for them in the confidence intervals, again remain as challenges.

**Acknowledgements** We thank Doug Oard, William Webber, Jason Baron and the two anonymous reviewers for their helpful remarks on drafts of this chapter. Also, we would like to thank Jason Baron, Doug Oard, Ian Soboroff and Ellen Voorhees for their support and advice in undertaking the various challenges of measuring effectiveness in the TREC Legal Track, and also all of the track contributors and participants without whom the track would not have been possible.

## References

1. Allan J, Carterette B, Dachev B et al (2008) Million query track 2007 overview. In: Proceedings of TREC 2007. <http://trec.nist.gov/pubs/trec16/papers/1MQ.OVERVIEW16.pdf>
2. Baron JR (ed) (2007) The Sedona conference<sup>®</sup> best practices commentary on the use of search and information retrieval methods in e-discovery. Sedona Conf J VIII:189–223
3. Baron JR, Lewis DD, Oard DW (2007) TREC-2006 legal track overview. In: Proceedings of TREC 2006. <http://trec.nist.gov/pubs/trec15/papers/LEGAL06.OVERVIEW.pdf>
4. Buckley C, Dimmick D, Soboroff I, Voorhees E (2006) Bias and the limits of pooling. In: SIGIR 2006, pp 619–620
5. Büttcher S, Clarke CLA, Soboroff I (2007) The TREC 2006 terabyte track. In: Proceedings of TREC 2006. <http://trec.nist.gov/pubs/trec15/papers/TERA06.OVERVIEW.pdf>
6. Carterette B, Soboroff I (2010) The effect of assessor errors on IR system evaluation. In: SIGIR 2010, pp 539–546
7. Harman DK (2005) The TREC test collections. In: TREC: Experiment and evaluation in information retrieval, pp 21–52
8. Hedin B, Tomlinson S, Baron JR, Oard DW (2010) Overview of the TREC 2009 legal track. In: Proceedings of TREC 2009. <http://trec-legal.umiaccs.umd.edu/LegalOverview09.pdf>
9. Lewis D, Agam G, Argamon S et al (2006) Building a test collection for complex document information processing. In: SIGIR 2006, pp 665–666
10. Oard DW, Baron JR, Hedin B et al (2010) Evaluation of information retrieval for e-discovery. *Artif Intell Law*
11. Oard DW, Hedin B, Tomlinson S, Baron JR (2009) Overview of the TREC 2008 legal track. In: Proceedings of TREC 2008. <http://trec.nist.gov/pubs/trec17/papers/LEGAL.OVERVIEW08.pdf>
12. Thompson SK (2002) *Sampling*, 2nd edn. Wiley, New York
13. Tomlinson S (2007) Experiments with the negotiated boolean queries of the TREC 2006 legal discovery track. In: Proceedings of TREC 2006. <http://trec.nist.gov/pubs/trec15/papers/opentext.legal.final.pdf>
14. Tomlinson S (2008) Experiments with the negotiated boolean queries of the TREC 2007 legal discovery track. In: Proceedings of TREC 2007. <http://trec.nist.gov/pubs/trec16/papers/open-text.legal.final.pdf>
15. Tomlinson S (2009) Experiments with the negotiated boolean queries of the TREC 2008 legal track. In: Proceedings of TREC 2008. <http://trec.nist.gov/pubs/trec17/papers/open-text.legal.rev.pdf>
16. Tomlinson S, Oard DW, Baron JR, Thompson P (2008) Overview of the TREC 2007 legal track. In: Proceedings of TREC 2007. <http://trec.nist.gov/pubs/trec16/papers/LEGAL.OVERVIEW16.pdf>
17. van Rijsbergen CJ (1979) *Information retrieval*, 2nd edn. Butterworths, London. <http://www.dcs.gla.ac.uk/Keith/Preface.html>
18. Voorhees EM (2000) Variations in relevance judgments and the measurement of retrieval effectiveness. *Inf Process Manag* 36(5):697–716
19. Voorhees EM, Harman D (1997) Overview of the fifth text retrieval conference (TREC-5). In: Proceedings of TREC-5. <http://trec.nist.gov/pubs/trec5/papers/overview.ps.gz>
20. Webber W (2010) Accurate recall confidence intervals for stratified sampling. Manuscript
21. Webber W, Oard DW, Scholer F, Hedin B (2010) Assessor error in stratified evaluation. In: CIKM 2010, pp 539–548
22. Yilmaz E, Aslam JA (2006) Estimating average precision with incomplete and imperfect judgments. In: CIKM 2006, pp 102–111
23. Zobel J (1998) How reliable are the results of large-scale information retrieval experiments. In: SIGIR 1998, pp 307–314

# Chapter 9

## Large-Scale Logical Retrieval: Technology for Semantic Modelling of Patent Search

Hany Azzam, Iraklis A. Klampanos, and Thomas Roelleke

**Abstract** Patent retrieval has emerged as an important application of information retrieval (IR). It is considered to be a complex search task because patent search requires an extended chain of reasoning beyond basic document retrieval. As logic-based IR is capable of modelling both document retrieval and decision-making, it can be seen as a suitable framework for modelling patent data and search strategies. In particular, we demonstrate logic-based modelling for semantic data in patent documents and retrieval strategies which are tailored to patent search and exploit more than just the text in the documents. Given the expressiveness of logic-based IR, however, there is an attendant compromise on issues of scalability and quality. To address these trade-offs we suggest how a parallelised architecture can ensure that logical IR scales in spite of its expressiveness.

### 9.1 Introduction

Patent retrieval has recently emerged as an important application of information retrieval (IR) and related research disciplines [13]. It differs from other IR applications, such as the Web, in many ways. A typical intellectual property (IP) professional searcher is an expert user in specialised software, which typically runs against a relational database of patents. The queries issued are usually very long and are often expressed in Boolean logic. The patents themselves are typically long and typographically sound. However, parts are occasionally written with the intention not to be found or read; such instances directly contradict principal assumptions of IR. Most importantly, patent documents contain a large number of semantic data

---

H. Azzam (✉) · T. Roelleke  
Queen Mary University of London, London, UK  
e-mail: [hany@eecs.qmul.ac.uk](mailto:hany@eecs.qmul.ac.uk)

T. Roelleke  
e-mail: [thor@eecs.qmul.ac.uk](mailto:thor@eecs.qmul.ac.uk)

I.A. Klampanos  
University of Glasgow, Glasgow, UK  
e-mail: [iraklis@dcs.gla.ac.uk](mailto:iraklis@dcs.gla.ac.uk)

such as entities (e.g. inventors and assignees) and relationships (e.g. worksFor and inventedBy). This data can be found implicitly within the text or explicated using, for example, markup languages such as XML. Naturally, such data can be used to return answers tailored to the searcher's need (e.g. an inventor's claims) rather than full patent documents. More generally, using this data leads to "semantic-aware" retrieval methods (retrieval strategies) and queries that exploit far more than *just* the text.

To make use of the semantics in patent documents and to deal effectively with the inherent properties of patent searching, an alternative way to model information needs and patent data is required. To this end, we propose a probabilistic and logic-based framework that combines databases and information retrieval (DB+IR). Adopting such a framework allows us to describe novel information tasks as well as traditional IR models in probabilistic Datalog or probabilistic SQL (PSQL). However, this modelling flexibility comes at a price with regard to efficiency and scalability. The underlying data structures that support logic-based retrieval against a DB+IR description of the information practically scale up to only a few thousand patent documents [4]. We address this scalability issue through distribution and parallelisation.

This chapter contributes a useful and flexible logic-based approach for modelling patent data and retrieval strategies that are tailored to semantic-aware patent search. Moreover, it proposes a parallel three-tier architecture to ensure that the logic-based approach remains scalable. Overall, this chapter demonstrates how the expressiveness and the processing of logic-based retrieval scales with respect to task complexity and enables knowledge engineers to solve complex search tasks, while remaining in control of the recording, repeating and adaptation of search strategies.

The remainder of this chapter is organised as follows: Section 9.2 motivates logical retrieval and presents the background. Section 9.3 contributes the logical modelling of patent search, of which we address two dimensions: the modelling of *data* and the modelling of *strategies* (retrieval scenario). Then, Section 9.4 gives an overview of how logical retrieval can be utilised for the descriptive modelling of distributed IR (DIR). This leads to an architecture scalable with respect to data volume and processing time requirements. Section 9.5 concludes the discussion and outlines the future work.

## 9.2 Motivation and Background

### 9.2.1 Logic-Based Retrieval

By "logic-based retrieval", we refer to an approach that applies a logic-based technology to retrieval tasks. Inherent to logic-based technology is the use of structured languages, such as probabilistic flavours of SQL and Datalog, to represent knowledge or information and queries. Due to the expressiveness of such languages, logic-based technology has a higher expressive power than the bag-of-words approach of traditional free-text-oriented IR.

Moreover, logic-based retrieval inherently supports reasoning about words and documents as well as about other modelled entities, depending on the application at hand. Logical languages allow for the expression of strategic and analytical queries, for instance, “Find the patents of Kraft and Cadbury and compare their most important patents. From these deduce the most dominant inventors in the food provision market.”, etc.

The application of logical retrieval to large-scale, partly poorly structured data, is challenging for a number of reasons. First, because of its expressiveness, logical retrieval demands an algebraic evaluation that is significantly more complex than fetching document IDs from the posting list of an inverted file. Additionally, the modelling of selection, ranking, and fusion strategies in a logical language is also challenging. However, the transparent, high-level abstraction of logic-based approaches is precisely what satisfies complex search requirements as they occur, in this context, in the area of patent searching.

Specifically in the case of patent searching, the properties of logical retrieval successfully meet the application requirements in the sense that patent searching is a complex retrieval task that requires reasoning about objects. Furthermore, it also requires modelling IR in such a way that users (IP professional searchers) can understand and modify the ranking and the reasoning processes. From personal communication with IP professionals, we understand that being in a position to understand and modify the underlying information description and retrieval mechanisms is essential for generating the trust needed in order to move on from Boolean retrieval. In other words, they require a sense of being fully in control of the retrieval and related processes. Due to their transparency and power of abstraction, logic-based approaches satisfy this requirement.

### 9.2.2 Probabilistic Datalog

Probabilistic Datalog (PDatalog) [2, 8] is a language for probabilistic reasoning, where deterministic Datalog has its roots in deductive databases. PDatalog was extended in [9, 14] improving its expressiveness and scalability for modelling IR models (ranking functions). Other logic-based languages that attach probabilities to logical formulae include PHA [6], PRISM [10] and MLNs [7]. In this chapter we use PDatalog in order to demonstrate the modelling of source selection and result fusion and, as a consequence, the modelling of retrieval strategies to support complex retrieval tasks, such as patent search. Figure 9.1 demonstrates an extract of PDatalog syntax.

Specific to the PDatalog shown here (based on [9]), as apposed to the PDatalog introduced in [3], is the specification of probability aggregation and estimation assumptions in goals and subgoals. We refer to the assumption between a predicate name and an argument list as an *aggregation* assumption. For example, for disjoint events, the sum of probabilities is the resulting tuple probability. The assumptions DISJOINT and SUM are synonyms. Alternatively, a subgoal may be specified with an *estimation* assumption. For example, for disjoint events, the sub-

<p>pdRule ::= head ':' body  pdDef ::= head ':=' body  head ::= goal  body ::= subgoalList  subgoalList ::= subgoal {'&amp;' subgoalList}</p>
<p>goal ::= tradGoal   aggGoal   estGoal  tradGoal ::= NAME '(' argList ')'  aggGoal ::= NAME aggAssumption '(' argList ')'  estGoal ::= NAME '(' argList ') ' ' {estAssumption} evidenceKey  subgoal ::= goal</p>
<p>argList ::= arg {',' argList}  varList ::= var {',' varList}  var ::= VARIABLE  arg ::= VARIABLE   constant  constant ::= STRING   NAME   NUMBER  evidenceKey ::= '(' varList ')'</p>
<p>tradAssumption ::= 'DISJOINT'   'INDEPENDENT'   'SUBSUMED'  logAssumption ::= 'SUM_LOG'   'MAX_LOG'  complexAssumption ::= 'DF'   'MAX_IDF'   'TF'   'MAX_ITF'   ...  aggAssumption ::= tradAssumption   'SUM'   'MAX'   ...  estAssumption ::= tradAssumption   logAssumption   complexAssumption</p>

**Fig. 9.1** PDatalog Syntax: Expressions between curly brackets, '{' and '}', are optional; a rule consists of a head and a body; a head is a goal, and a body is a subgoal list

goal “index(Term, Doc) | DISJOINT(Doc)” expresses the conditional probability  $P(\text{Term}|\text{Doc})$ , derived from the statistics in the relation “index”. Complex assumptions, such as DF (for document frequency) or MAX\_IDF (for maximum inverse document frequency), can be specified to describe probabilistic parameters commonly used in information retrieval in a convenient way. For further details see [9].

A PDatalog rule is evaluated such that the head is true if and only if the body is true. For example, the following rules demonstrate a common term matching strategy in IR. Queries are represented in the relation “qterm( $T$ ,  $Q$ )” (where  $T$  is a term and  $Q$  is a query ID), and the collection is represented in the relation “term( $T$ ,  $D$ )” (where  $D$  is a document ID). If  $T$  occurs in  $Q$ , and  $T$  can be found in  $D$ , then  $D$  is retrieved for  $Q$ .

<pre> 1 coord_match(D, Q, T) :- qterm(T, Q) &amp; term(T, D); 2 coord_retrieve SUM(D, Q) :- coord_match(D, Q, T); </pre>
--

In subsequent sections we demonstrate how such an approach can be used to model patent search and the DIR processes. We maintain that modelling such tasks in a modular and transparent way gives IP professionals, as well as to other expert users in general, flexibility and tractability of search sessions and predictable retrieval effectiveness.



### 9.2.3 Logical Modelling of Semantic Data

This section reviews the modelling layers that are used to build complex representations of patent documents and the underlying textual, structural and semantic data. These modelling layers are built upon object-oriented and relational modelling concepts. The first of these layers is illustrated next.

#### 9.2.3.1 The Probabilistic Object-Oriented Content Model

The Probabilistic Object-Oriented Content Model (POOCM) joins

- concepts of probability theory,
- concepts of object-oriented modelling (OOM), and
- concepts of content modelling (CM).

Probability theory comprises a range of concepts such as probability estimation and aggregation. OOM is based on relationships between objects and classes. CM is traditionally concerned with the keyword-based “index” of objects.

The combination leads to an approach to modelling where conventional concepts (monadic predicates in OOM) become concepts-in-a-context (dyadic predicates), and conventional roles (dyadic predicates in OOM) become roles-in-a-context. Moreover, there is a new type of predicate, referred to as “terms”. In a conventional model, these are zero-arity predicates; in the POOCM, they are monadic predicates (the context is the parameter). The following example illustrates the nature of the POOCM:

```

1 0.5 machine(doc_6296192); #term predicate
2 0.7 inventor(hecht_david, doc_6296192); #classification predicate
3 0.4 worksFor(hecht_david, xerox_corp, doc_6296192); #relationship predicate
4 pubdate(doc_6296192, 20011002, patent_database); #attribute predicate
```

To implement the POOCM, we utilise an object-relational approach. Alternatively, one could imagine a POOCM-native approach, but for the focus of this chapter, we pursue the object-relational route, which is illustrated next.

#### 9.2.3.2 The Probabilistic Object Relational Content Modelling (PORCM)

The Probabilistic Object-Relational Content Model (PORCM) is the base for implementing the POOCM. Additionally, it is the data model used to represent patent data. The PORCM combines:

- concepts of probability theory,
- concepts of object-relational modelling (ORM), and
- concepts of content modelling (CM).

**Fig. 9.2** The probabilistic object-relational content model

Schema
relship(RelshipName, Subject, Object, Context)
attribute(AttrName, Object, Value, Context)
classification(ClassName, Object, Context)
term(Term, Context)
is_a(SubClass, SuperClass, Context)

Object-relational modelling (see [11] and [12]) utilises relations to represent the concepts of OOM. In principle, the PORCM can be viewed as the relational implementation of the POOCM. Therefore, object-oriented concepts (i.e. relationships between objects, classification of objects) are modelled in a relational schema and a relation “term” and one attribute (“context”) are added to support content-oriented modelling. The resulting schema is illustrated in Fig. 9.2.

In this schema each relation represents a component of the POOCM. Specifically, there are relations for subject-object and object-value relationships. One could view constant values (strings, numbers, names) as objects, or model object-object and object-value relationships in different relations. We chose the latter option. The relation “attribute(Name, Object, Value, Context)” is for object-value associations, and the relation “relship(Name, Subject, Object, Context)” is used for subject-object associations. For object-class associations we use the relation “classification(ClassName, Object, Context)”.

There is also a relation for generalisation. Generalisation (class hierarchy) is a relationship between classes. Similar to the way relationships and attributes are kept separate, generalisation is modelled in a relation such as “is\_a(SubClass, SuperClass, Context)”. Since class hierarchy can also be modelled via rules, it is a modelling choice whether to instantiate class relations or to model generalisation using rules. In this chapter we opt for the latter.

Having introduced the POOCM and its relational implementation (the PORCM), we now discuss how such logic-based modelling can be scaled.

### 9.3 Logical Modelling of Patent Search

The logical modelling of patent search builds upon

1. the logical modelling of semantic data, and
2. the logical modelling of retrieval strategies.

The logical modelling of semantic data and ranking strategies has the following main benefit:

Strategies can be recorded, exchanged, repeated and refined.

The descriptive and logic-based modelling approach as opposed to black-box search systems provides patent searches with a transparent technology. This technology can

```

1 <XML>
2 <PATENTS>
3 <DOC id="6296192">
4 <PAT-NO>6296192</PAT-NO>
5 <APP-NO>465990</APP-NO>
6 <APP-DATE>19991216</APP-DATE>
7 <PAT-TYPE>1</PAT-TYPE>
8 ...
9 <PUB-DATE>20011002</PUB-DATE>
10 <PRI-IPC>G06K 19/06</PRI-IPC>
11 <PRI-USPC>235494</PRI-USPC>
12 <INVENTOR id="hecht_david">Hecht; David L.</INVENTOR>
13 <ASSIGNEE id="xerox_corp">Xerox Corporation</ASSIGNEE>
14 <TITLE>Machine-readable record with a two-dimensional lattice ....</TITLE>
15 <ABST>A machine-readable record is provided for ....</ABST>
16 <SPEC>DETAILED DESCRIPTION OF THE ILLUSTRATED ... </SPEC>
17 <CLAIM>What is claimed: 1. A machine-readable record for ...
18 2. The record of claim 1 wherein said first and second directions are
19 lines orthogonal to each other.
20 3. The record of claim 1 wherein said first and second sequences are ...
21 4. The record of claim 1 wherein said synchronizing code comprises
22 glyphs having ....
23 </CLAIM>
24 </DOC>
25 </PATENTS>
26 </XML>

```

**Fig. 9.3** An XML-based representation of a patent document

help the searchers to maintain high standards for compliance (issuing repeatable strategies/searches and understanding how and why was the result obtained) when performing patent-related search tasks.

Furthermore, the logical modelling of retrieval strategies separates in a clear manner what we are looking for, in which context and with which task in mind, from other components (modules), such as the ranking strategy. For example, the ranking strategy (e.g. TF-IDF, BM25, LM) can be easily exchanged while other components, such as the post-processing of the results, remains the same.

### 9.3.1 Representing Patent Documents Using the PORCM

Figure 9.3 demonstrates an example of a patent document represented in eXtensible Markup Language (XML). The example, in particular, consists of XML elements, which can be classified into two broad categories based on the nature of the encap-

sulated content: logical elements, such as section, image and title, which define the structure of the document; and semantic elements (semantic structures), such as inventor and assignee that label the contents of a text. The latter are analogous to a database schema that represents the important semantics of the data rather than its structural layout.

Figure 9.4 shows how the PORCM models the aforementioned XML representation (for the purpose of this chapter we assume that there is an XML parser that extracts the shown relations). The term-based relations (“term” and “term\_doc”) model term-oriented representations, which are common in traditional IR. The “term” relation stores the parsed text and the *context* where the text was found. The context is a general concept that refers to documents, sections, databases and any other object with a content. In this figure, the context is expressed in XPath.<sup>1</sup> The relation “term\_doc” is derived from the “term” relation. This relation maintains only the root context (the document) of each term-element pair, which propagates the content knowledge found in the children context to the root context.

The classification, relationship and attribute relations represent object-class, subject-object and object-value associations, respectively. Note that from the figure we can identify two types of objects: *structural objects*, such as the title and claim elements where the XPath expressions are used as the objects’ IDs; and *semantic objects*, which are identified using semantic IDs, such as “hecht\_david”. From a conceptual point of view, classifications, relationships and attributes of *all* objects are maintained together (i.e. all classifications in one relation, all relationships in one relation and all attributes in one relation). From a pragmatic, engineering-oriented point of view, it makes often sense to keep structural elements and semantic objects separated. This separation helps to efficiently reason across both types of objects. For example, there is no need to model relationships between structural objects. Additionally, for XPath-like object IDs, the classification may be inferred from the object Id, and hence, the classification of structured objects does not need to be materialised. This reduces disc usage and minimises the number of indices to be maintained.

The following example illustrates how we can discern structural from semantic object IDs and structural from semantic relations using the PORCM and PDataLog.

```

1 # Structural predicates derived from basic predicates:
2 inventorElement(XPath, Context) :-
3     classification(inventor, XPath, Context);
4 id(XPath, Value, Context) :-
5     attribute(id, XPath, Value, Context);

7 # Semantic classifications derived from structural and basic predicates:
8 inventorEntity(Id, Context) :-
9     inventorElement(XPath, Context) & id(XPath, Id, Context);

```

<sup>1</sup><http://www.w3.org/TR/xpath>.

term	
Term	Context
...	...
19991216	6296192/app-date[1]
...	...
hecht	6296192/inventor[1]
david	6296192/inventor[1]
xerox	6296192/assignee[1]
corporation	6296192/assignee[1]
...	...
machine	6296192/title[1]
readable	6296192/title[1]
record	6296192/title[1]
...	...

(a) Term proposition in the element contexts

term_doc	
Term	Context
19991216	6296192
...	...
hecht	6296192
david	6296192
xerox	6296192
corporation	6296192
...	...
machine	6296192
readable	6296192
record	6296192
...	...

(b) Term proposition in the root contexts

classification		
ClassName	Object	Context
...	...	...
inventor	6296192/inventor[1]	6296192
assignee	6296192/assignee[1]	6296192
title	6296192/title[1]	6296192
...	...	...
claim	6296192/claim[1]	6296192
...	...	...

(c) Classification proposition

relationship			
RelName	Subject	Object	Context
worksFor	hecht_david	xerox_corp	patent_db
inventedBy	6296192	hecht_david	patent_db
...	...	...	...

(d) Relationship proposition

attribute			
AttrName	Object	Value	Context
id	6296192/inventor[1]	“hecht_david”	6296192
id	6296192/assignee[1]	“xerox_corp”	6296192
...	...	...	...

(e) Attribute proposition

Fig. 9.4 A PORCM-based representation of a patent document

The first two rules derive structural predicates from the basic PORCM layer, and the structural object IDs are made explicit. The rules, in particular, “lift” the attributes to become relation names, resulting in basic classifications and attributes becoming structural classifications and attributes.

The third rule extracts a semantic object (a name entity) by combining structural information about elements of type “inventor” and their attributes. This type of modelling of entities, in particular, is prevalent in Entity-Relationship-graphs, such as RDF, where URIs are used to denote objects. In the derived entity, the “Id”, thus, denotes a unique identifier that corresponds to the attribute value “hecht\_david” in Fig. 9.4.

This example, therefore, highlights the different types of layers that can be derived from the PORCM, namely basic, structural and semantic. These layers help to achieve textual and (semi-)structured data independence, which is analogous to the notion of “data independence” from the classic ANSI SPARC standard. Any data (e.g. XML, RDF, RSS) can be represented in the application-independent the PORCM, after which patent-specific relations are derived.

Another advantage is that a precise and tidy schema design helps to achieve reusable and transferrable implementations. For example, indexing and processing strategies developed for one type of patent retrieval application (experiments) become transferrable, provided that the structural and semantic layers are derived as previously shown.

Lastly, explicitly stating how the basic and the semantic layers are related impacts the modelling of probability estimations and aggregations required by retrieval models for patent search. The predicates in the basic PORCM can be used to construct an evidence space for term-based (e.g. TF-IDF, LM) retrieval models and for basic semantic models (e.g. attribute-based LM). In the derived structural and semantic layers, however, more complex and tailored (patent-specific) models can be constructed. For example, an LM-based ranking of inventors according to their experience can be easily constructed using an evidence space based on the classifications in the semantic layer of PORCM. We discuss next the components for modelling retrieval strategies for patent search and, more generally, probability estimations and aggregations leveraged by the aforementioned three schema layers.

### 9.3.2 Retrieval Scenario

This section presents a retrieval scenario that illustrates how logical modelling can be used to model patent-specific retrieval strategies. Let us consider that after an initial search for patents about semiconductors (the initial search being a basic search session), the task is to explore the patents of *experienced* inventors who filed the most relevant patents about semiconductors. Logical retrieval supports the required reasoning process as follows: (1) initial ranking strategies (TF-IDF, LM, BM25, boolean, coord-match) that retrieve patents about semiconductors; (2) post-processing of retrieved results to find structural elements and named entities (e.g. inventors); (3) and modelling of predicates, such as *experienced inventor*. Next we

illustrate how this reasoning process (retrieval strategy) can be expressed in PData-log.

### 9.3.2.1 Initial Ranking Strategy Using TF-IDF

We chose TF-IDF as the initial ranking strategy. The strategy is parameterised with three external relations: “qterm”, “pidf”, and “tf\_d”. The relation “qterm(Term, Query)” contains the query terms, which in this case are the terms “semi” and “conductors”. The relation “tf\_d(Term, Context)” contains term-document pairs with tuple probabilities  $P(t|d)$ , which are proportional to term frequency, and “pidf(T)” (probabilistic IDF) contains terms where the tuple probabilities are proportional to the IDF-values of the respective terms. These probabilities have been estimated using the “DISJOINT” and “MAX\_IDF” operators, respectively (see Sect. 9.2). The probabilistic assumption, “SUM”, specifies how the probabilities (weights) of the non-distinct tuples are to be aggregated.

```

1 # TF-IDF
3 # IDF-based query term weighting:
4 qterm_pidf(Term, Query) :- qterm(Term, Query) & pidf(Term);
6 # Normalisation:
7 w_qterm(Term, Query) :- qterm_pidf(Term, Query) | (Query);
9 # Match over terms:
10 match(Context, Query, Term) :- w_qterm(Term, Query) & tf_d(Term, Context);
12 # Retrieval (aggregation of evidence from match):
13 retrieve SUM(Context, Query) :- match(Context, Query, Term);

```

### 9.3.2.2 Post-processing of Retrieved Results

The retrieval strategy extracts the inventors of the retrieved patent documents. Additionally, it derives the semantic IDs of the inventors from the structural inventor elements and the basic predicates. Below we demonstrate how this post-processing step is implemented.

```

1 # TF-IDF ranking of patents about semiconductors
2 retrievedPatents(Context, Query) :- retrieve(Context, Query);
4 # Structural predicates derived from basic predicates
5 inventorElement(XPath, Context) :- classification(inventor, XPath, Context);
6 id(XPath, Value, Context) :- attribute(id, XPath, Value, Context);
8 # Select the inventor elements that occur in the retrieved patents

```

```

9 inventorsOfRetrieved(XPath, Context) :-
10   retrievedPatents(Context, Query) & inventorElement(XPath, Context);
12 # Semantic classifications derived from structural and basic predicates:
13 inventorEntity(Id, Context) :-
14   inventorsOfRetrieved(XPath, Context) & id(XPath, Id, Context);

```

Note that the ranking strategy (TF-IDF) can be easily replaced with another strategy such as BM25 or LM without affecting the implementation of the post-processing step. This modularity is one of the main features of a logical modelling approach.

### 9.3.2.3 Modelling of Experienced Inventor

Using logical modelling we can also model “vague” (probabilistic) predicates such as “experience” and “popularity”. The example below demonstrates how this modelling is performed.

```

1 # Retrieve only the experienced inventors
2 experiencedInventors SUM (Id) :- inventorEntity(Id, Context) | DISJOINT();

```

Using the “inventorEntity”, which consists of a non-distinct list of inventors’ IDs and their contexts (patents), we perform a frequency-based estimate to find out the experienced inventors. The intuition behind this estimate is that the more patents has a particular inventor, the more experienced he/she is. Note that the experienced inventors are mainly in the “semiconductors” domains since “inventorEntity” is derived from the retrieved inventors. A more general approach would be to model “experience” using an external knowledge base containing the inventors and the number of their inventions/patents.

### 9.3.2.4 Exploring the Patents of the Experienced Inventors

We can find the patents of experienced inventors in “semiconductors” using the list of experienced inventors. We join the “experiencedInventors” with a “patentDB” (a database of patents and their inventors), which results in a list of patents about “semiconductors” that were issued by experienced inventors.

```

1 #Retrieve patents of experienced inventors
2 patentsOfExperiencedInventors (Context):-
3   experiencedInventors (Id) & patentDB(Id, Context);

```

Given the trade-off between expressiveness and scalability, we demonstrate how a parallelised architecture can ensure that the logical modelling approach scales.



## 9.4 Parallelising the Workload

The transparency and the power of abstraction offered by the logic-based modelling of semantic patent searching is costly with regard to both space and processing requirements. A way to alleviate this problem without sacrificing the expressiveness and transparency of logic-based DB+IR is through distribution or parallelisation, depending on the deployment infrastructure at hand. Logic-based DB+IR itself allows for expressing Distributed Information Retrieval (DIR, [1]) components, such as *resource description*, *resource selection* and *fusion of results* in a unified and transparent way. The technology developed as part of the Large-Scale Logical Retrieval (LSLR) project [4, 5] can form the basis upon which logic-based semantic searching is offered.

For an example of how the process of source selection can be expressed within the logic-based framework, consider the following rules, expressed in PDataLog. Here, “Src” corresponds to an individual knowledge base (KB). Worth-noting is the parallel between “Src” below and “Context” in the modelling of TF-IDF ranking of the previous section.

```

1 # Generate representation of sources:
2 df_src(Term, Src) :- tf_src(Term, Doc, Src);

4 # Match over terms:
5 match(Src, Query, Term) :- w_qterm(Term, Query) & df_src(Term, Src);

7 # Selection (aggregate evidence from match):
8 select_src SUM(Src, Query) :- match(Src, Query, Term);

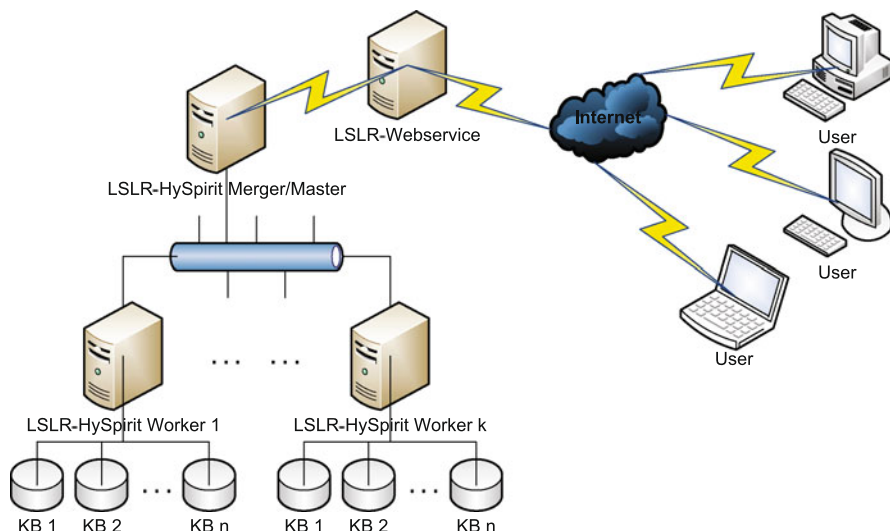
```

The example also demonstrates how the same framework that is used to model the retrieval strategies is also utilised to model source selection. This emphasises the flexibility and openness of the logic-based approach.

The overall LSLR architecture can be seen in Fig. 9.5. In this architecture, each LSLR-worker processes each query by first ranking the knowledge bases with regard to how appropriate they are to answer it. It then forwards the query to the top- $k$  KBs. During this phase, and because of our transparent logic-based approach, the set of selected KBs are essentially treated as a single collection, therefore doing away with the need to explicitly merge the results. Here, the more KBs selected, the more time an LSLR-worker needs to reply to a query. Each LSLR-worker can handle multiple KBs depending on the semantics we want to be able to process at run-time.

On the level of the LSLR-HySpirit-Master, when a query is received, it is forwarded to all connected LSLR-HySpirit-Workers. Even though the LSLR-Master is in a position to set out retrieval and selection strategies for individual LSLR-HySpirit-Workers to follow, it is effectively unaware of their overall content. Therefore, given a query, all LSLR-HySpirit-Workers will be working in parallel, while restricting individually that is their searching to a subset of their KBs.

Strategies that utilise the partial KBs on the local nodes in order to provide source selection and results’ fusion can be provided in a consistent and transparent way



**Fig. 9.5** An overview of the LSLR architecture

syntactically similar to patent searching. The exact design of LSLR and the logical modelling of distributed and parallel IR processes is not within the focus of this chapter. More details can be found in [4, 5].

## 9.5 Conclusions and Future Work

The transparency and power of abstraction that logic-based modelling offers makes it a suitable framework for a wide variety of search applications. Sophisticated, non-ad-hoc search tasks involving expert users, such as patent searching, can greatly benefit from its adoption. DB+IR systems implement the principles of logic-based retrieval through the use of structured probabilistic languages, such as probabilistic Datalog, which is used in this chapter. For these target application domains, the use of such languages allows for modular, transparent and tractable designs which are decoupled from the underlying data or information models employed. In particular, such languages can be leveraged to model the semantic data found in patent documents, resulting in more sophisticated retrieval methods (retrieval strategies) and queries. However, the scalability of such modelling techniques—with regard to data volume and processing times—becomes a significant challenge due to the increase in computational complexity of high-level abstraction and expressiveness.

In this chapter, we primarily examined the “scalable” expressiveness of logical retrieval by modelling data *and* strategies (uncertain reasoning) that can potentially solve complex (semantic) retrieval tasks. We also discussed the usage of distribution and parallelisation. After introducing the basics of probabilistic Datalog, we modelled distributed IR techniques such as source selection and fusion, and so

demonstrated that our approach is both modular and expressive. Attending to the requirements of semantic-aware patent search, we proposed a three-tier distributed architecture that is able to adapt to different levels of hardware resource provision by maximising concurrency.

Probabilistic logical modelling can be viewed as a “protocol language” that allows web services to transfer and exchange query representations and retrieval strategies. Future work includes service selection and traffic balancing as well as the creation of further content distributions, parallelisation strategies and resource selection algorithms.

**Acknowledgements** We would like to thank Matrixware Information Services GmbH and the Information Retrieval Facility (IRF) for supporting this work. We also would like to thank Helmut Berger for his management of the LSLR project. Finally, many thanks to the reviewers for their excellent suggestions.

## References

1. Callan J (2000) Distributed information retrieval. In: *Advances in information retrieval*. Kluwer Academic, Dordrecht, pp 127–150
2. Fuhr N (1995) Probabilistic datalog—a logic for powerful retrieval methods. In: Fox E, Ingwersen P, Fidel R (eds) *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval*. ACM, New York, pp 282–290
3. Fuhr N (1996) Optimum database selection in networked ir. In: *SIGIR workshop on networked information retrieval*
4. Klampanos IA, Azzam H, Roelleke T (2009) A case for probabilistic logic for scalable patent retrieval. In: *CIKM workshop on patent information retrieval*, pp 1–8
5. Klampanos IA, Wu H, Roelleke T, Azzam H (2010) Logic-based retrieval: Technology for content-oriented and analytical querying of patent data. In: *IRFC*, pp 100–119
6. Poole D (1993) Probabilistic horn abduction and Bayesian networks. *Artif Intell* 64(1):81–129
7. Richardson M, Domingos P (2006) Markov logic networks. *Mach Learn* 62(1–2):107–136
8. Roelleke T, Fuhr N (1998) Information retrieval with probabilistic datalog. In: Crestani F, Lalmas M, Rijsbergen CJ (eds) *Uncertainty and logics—advanced models for the representation and retrieval of information*. Kluwer Academic, Dordrecht
9. Roelleke T, Wu H, Wang J, Azzam H (2008) Modelling retrieval models in a probabilistic relational algebra with a new operator: The Relational Bayes. *VLDB J* 17(1):5–37
10. Sato T, Kameya Y (2001) Parameter learning of logic programs for symbolic-statistical modeling. *J Artif Intell Res* 15:391–454
11. Scholl M, Schek HJ (1990) A relational object model. In: Abiteboul S, Kanellakis P (eds) *ICDT '90*. Springer, Berlin, pp 89–105
12. Stonebraker M, Moore D, Brown P (1998) *Object-relational DBMSs: Tracking the next great wave*. Morgan Kaufmann, San Francisco
13. Tait J, Lupu M, Berger H, Roda G, Dittenbach M, Pesenhofer A, Graf E, van Rijsbergen K (2009) Patent search: An important new test bed for ir. In: Aly R, Hauff C, den Hamer I, Hiemstra D, Huibers T, de Jong F (eds) *9th Dutch–Belgian information retrieval workshop (DIR 2009)*, TNO ICT, Delft, The Netherlands. Neslia Paniculata, Enschede
14. Wu H, Kazai G, Roelleke T (2008) Modelling anchor text retrieval in book search based on back-of-book index. In: *SIGIR workshop on focused retrieval*, pp 51–58

# Chapter 10

## Patent Claim Decomposition for Improved Information Extraction

Peter Parapatics and Michael Dittenbach

**Abstract** In several application domains research in natural language processing and information extraction has spawned valuable tools that support humans in structuring, aggregating and managing large amounts of information available as text. Patent claims, although subject to a number of rigid constraints and therefore forced into foreseeable structures, are written in a language even good parsing algorithms tend to fail miserably at. This is primarily caused by long and complex sentences that are a concatenation of a multitude of descriptive elements. We present an approach to split patent claims into several parts in order to improve parsing performance for further automatic processing.

### 10.1 Introduction

The claims in a patent can be seen as its essence, because they legally define the scope of the invention while the description and drawings have a supporting role to make the invention described more comprehensible. Both, the European<sup>1</sup> as well as the US definition<sup>2</sup> of patent claims put emphasis on conciseness and clarity. This and further official guidelines on claim formulation have several implications on the language used. In this work, we investigate how the structure of the claims-specific language can be used to split them into several components and rearrange them in order to improve the performance of natural language processing tools such as dependency parsers and to improve readability. To this end, we use the English language parts of a set of European patent documents from the International Patent

---

<sup>1</sup><http://www.epo.org/patents/law/legal-texts/html/epc/2000/e/ar84.html>.

<sup>2</sup><http://www.gpoaccess.gov/uscode/browse.html>.

P. Parapatics (✉)

Department of Software Technology and Interactive Systems, Vienna University of Technology,  
Favoritenstr. 9-11/188, 1040 Vienna, Austria  
e-mail: [p.parapatics@gmail.com](mailto:p.parapatics@gmail.com)

M. Dittenbach

max-recall information systems, Vienna, Austria  
e-mail: [m.dittenbach@max-recall.com](mailto:m.dittenbach@max-recall.com)

Classification (IPC) category A61C (Dentistry; Oral or Dental Hygiene). The goal of this research is the development of a method to automatically decompose the often long and winding sentences into smaller parts, identifying their constituents and relations and putting them into a machine-processable structure for further analysis and visualization.

## 10.2 Patent Claim Structure

In general, rules for examining, and thus also for drafting a patent are quite similar internationally, but there are variations from patent office to patent office. The characteristics described in this paper are based on the Guidelines for Examination in the European Patent Office (EPO) as of April 2009, Part C, Chap. III [2] and the Manual of Patent Examining Procedure of the United States Patent and Trademark Office (USPTO) [4]. The EPO as well as the USPTO require every patent document to contain one or more claims. The claims section is the only part of a patent conferring protection to the patent holder. The description and drawings should help the examiner to understand and interpret the claim but do not provide any protection themselves. Due to the importance of the claims there are very precise syntactic and semantic rules that have to be followed when drafting patent claims. A patent contains one or more independent claims that define the scope of the invention [2, Sect. 3.4]. Additionally, a patent may contain dependent claims which impose further limitations and restrictions on other dependent or independent claims. Each claim has to be written in a single sentence.

Independent claims should start with a part which describes already existing prior art knowledge and is used to indicate the general technical class of the invention. It describes the elements or steps of the invention that are conventional or known. These are then refined in a part describing the aspects or steps of the invention which are considered new or improved and which the patent holder wants to protect. These two parts are connected with specific key phrases which vary between the USPTO and EPO. Moreover, the terminology for naming the parts differs slightly. The USPTO refers to the part describing prior art as *preamble* [4, Chap. 608.01(i)]. The key phrase is called *transitional phrase* and the main part of the claim is referred to as the *claim body*. In the transitional phrase, keywords such as “comprises”, “including” or “composed of” are used. The EPO suggests the same claim structure but does not name the separate parts [2, Sect. 2.2]. It refers to this structure as the *two-part form* (not counting the transitional phrase) with the first part corresponding to the preamble and the second part to the claim body. The two parts are linked with either the phrase “characterized by” or “characterized in that”.

Independent claims do not necessarily have to be defined in the two-part form. The EPO [2, Sect. 2.3] considers the two-part form inappropriate for claims which describe:

- the combination of known integers of equal status, the inventive step lying solely in the combination;

- the modification of, as distinct from addition to, a known chemical process e.g. by omitting one substance or substituting one substance for another;
- or a complex system of functionally inter-related parts, the inventive step concerning changes in several of these or in their inter-relationships.

An example claim for the third rule is the following claim taken from a patent document in the dentistry domain: “A dental restoration comprising an outer shader layer, an intermediate layer which is substantially hue and chroma free and translucent and an opaque substructure which has a specific chroma on the Munsell scale and a specific Munsell hue.”

A dependent claim can refer to independent as well as other dependent claims and are used to refine and describe additional details or parts of the invention. It has to incorporate all features from the claim it refers to and must not broaden the previous claim. The EPO suggests the following structure for dependent claims: The first part of the claim contains a reference to all claims it depends on, followed by the refinement or the definition of parts of the invention. The two-part form, where the two parts are linked with “characterized in that” or “characterized by”, is not required for dependent claims but is nevertheless very common. Other common link phrases between the two parts are “wherein” and “comprising” such as in the claims “The orthodontic bracket of claim 1 *wherein* said bracket is [...]” or “An apparatus usable for carrying out the method according to claim 1 or 2, *comprising* [...]”

The USPTO explicitly defines rules for the order of claims in the patent [4, Chap. 608.01(n)]. In the EPO guidelines the order is stated implicitly. Dependent claims have to be ordered from the least restrictive to the most restrictive. This is important from a machine processing point of view, in the sense that concepts or terms which are refined in a dependent claim have already been introduced in a preceding claim in the document.

Claims have a different form depending the type of invention they describe. It can be differentiated between claims to physical entities (product, apparatus) and claims to activities (process, use) [2, Sect. 3]. *Product* and *apparatus claims* normally have the following form: “An X, comprising a Y and a Z”. *Method claims* have a very similar form but instead of describing parts of a physical entity a sequence of steps are described. “A method for X comprising (the steps of) heating Y and cooling Z”. A *use claim* is usually written in the following form: “The use of X for the Z of Y”.

Several common grammatical structures can be found in patent claims. One that is commonly used in claims is an enumeration of several parts of prior art improvements or steps of a method. These enumerations occur in various syntactic forms like: “An M comprising an X, a Y and a Z” or “An M comprising: (a) an X, (b) a Y and (c) a Z”.

Since a claim should be as concise as possible (cf. [2, Sect. 4]), each term used in the claim must have a definite and unambiguous meaning. New concepts are introduced with an indefinite article (“a” or “an”). Subsequent uses of the same element are preceded by “the” or by “said”.

### 10.3 Related Work

Research is done in various fields of patent processing.

In [7] the authors aim to quantify three challenges in patent claim parsing: claim length, claim vocabulary and claim structure. Their experiments show that the average sentence length of claims is longer compared to general English sentences even if the claims are split on semicolons and not only on full stops. This results in more structural ambiguities in parses of long noun phrases. While the vocabulary is similar to normal English texts the authors show that the distribution of words does differ. The biggest challenge for syntactic parsing poses the sentence structure as claims consist of sequences of noun phrases rather than clauses.

The authors of [3] propose a technique for claim similarity analysis which could be used for building patent processing tools to support patent analysts. They compute a similarity score between two claims based on simple lexical matching and knowledge based semantic matching. The syntactic similarity measure is based on the number of nouns that occur in both claims. For semantic similarity a score is computed by comparing each noun from the first claim to all nouns from the second claim using WordNet [1]. The highest score is recorded. The final semantic similarity score for two claims is then calculated by summing up the semantic similarity score for each noun.

A complex and domain-specific NLP-based approach is used in [5]. It is claimed that the use of broad coverage statistical parsers like the Stanford Natural Language Parser<sup>3</sup> is not appropriate for the patent domain. Since they are trained on general language documents, the accuracy of these parsers suffers when used for parsing patent claims. The proposed parsing method relies on supertagging and uses a domain-specific shallow lexicon for annotating each lexeme with morphological, syntactic and semantic information. Semantic information consists of an ontological concept defining the word membership in a certain semantic class (Object, Process, etc.). In the supertagging procedure each word is annotated with several matching supertags. In the following disambiguation procedure, hand crafted rules are used to eliminate incorrect supertags. The central part of the method is the predicate lexicon which is used to create a predicate-argument structure by annotating each predicate with syntactic and semantic information. A grammar is used to fill each argument of a predicate with a matching chunked phrase (e.g.: NP, NP and NP) from the claim based on the syntactic and semantic information in the supertag.

In [8] patent claims are compared by computing a similarity measure for conceptual graphs extracted from the claims using a natural language parser. A conceptual graph  $G$  is a set of  $(C, R, U, lab)$  where  $C$  are the concept vertices,  $R$  the relation vertices,  $U$  a set of edges for each relation. A label from the set  $lab$  is assigned to every vertex in the graph. A specific domain ontology is used for the concept and relation vertices in the conceptual graph. The conceptual graphs are extracted from dependency relations created with the Stanford Parser. The developed method is intended to be used for infringement searches and in particular for tasks such as patent clustering, patent comparison and patent summarization.

---

<sup>3</sup><http://nlp.stanford.edu/software/lex-parser.shtml>.

**Table 10.1** Data sets: characteristics

Data Set	Claim type	Nr. claims	Nr. words	Avg. claim length
Analyzed Set	Ind. claims	159	20,321	127.81
	Dep. claims	862	28,794	33.40
Evaluation Set	Ind. claims	13,628	1,803,341	132.33
	Dep. claims	73,706	2,415,533	32.77

The authors of [6] focus on structural analysis of Japanese patent claims in order to create parsing methods for specific claim characteristics. They show that Japanese patent claims are very similar to European and US claims in the sense that a single sentence out of multiple sentences using specific keywords and relations. Six common relations (Procedure, Component, Elaboration, Feature, Precondition, Composition) are described which can be found in Japanese patents. These relations can be identified by cue phrases, for which a lexical analyzer is used in order to decompose a patent claim into several parts.

## 10.4 Data Set

For creating and evaluating our method, which will be described in the next section, two data sets from the IPC category A61C (Dentistry, Oral or Dental Hygiene) were used. A data set of 86 randomly selected patents was manually analyzed for creating the decomposition rules (Analyzed Set) and a larger set of 5,000 patents was used for evaluation (Evaluation Set). The Analyzed Set only consists of patents filed at the EPO while the Evaluation Set consists of 774 European patents and 4,226 US patents. The patents were sampled from the Matrixware Research Collection (MAREC) data set.<sup>4</sup> Table 10.1 shows the characteristics of the two data sets. The figures show that independent claims are more than three times as long as dependent claims.

Table 10.2 shows the success rate (coverage) of the Stanford parser applied to the claims. A successful parse in this context does not refer to the correctness of the parse tree but only indicates that the parser was able to produce a result. The coverage provides a good indication for the complexity of a text. The higher complexity of independent claims is therefore underlined by the high number of unsuccessful parses of independent claims as compared to dependent claims. It can be seen that the average number of successful parses is significantly higher for dependent claims than for independent claims. Additionally, the success rate of the parser decreases significantly when reducing the maximum amount of memory (JVM max. heap size). This is an important parameter, because of the memory requirements for constructing the large parse trees for the relatively long independent claims. An infor-

<sup>4</sup><http://ir-facility.org>.



**Table 10.2** Stanford parser success rate

Data Set	Claim type	JVM max. heap size	Successful parses	Failed parses	% of successful parses
Analyzed Set	Ind. claims	1000 MB	132	27	83.01%
		500 MB	89	70	55.97%
	Dep. claims	1000 MB	859	3	99.65%
		500 MB	848	14	98.38%
Evaluation Set	Ind. claims	1000 MB	10,671	2,957	78.30%
		500 MB	7,482	6,146	54.90%
	Dep. claims	1000 MB	73,427	279	99.62%
		500 MB	72,769	937	98.73%

mal evaluation of the parse trees indicates that the quality of the results is very low for the long and complex claim sentences.

## 10.5 Method

### 10.5.1 Preprocessing

Before a patent document is decomposed, a number of data preprocessing and cleaning steps are executed to normalize the claim text. In patent claims, references to images are enclosed in parentheses. Their representation can include numbers as well as letters and range from simple forms such as “(21)” or “(12b)” to more complex constructs like “(21b; 23; 25c)”. For our purpose, these image links are not processed and pose problems for the extraction rules. The following regular expression is used for finding and removing image links (but retaining mathematical and chemical formulas):

```
(\\(\\s*[0-9][0-9a-z,;\\s]*\\))
```

In some claims, elements of an invention are enumerated in a form such as “a.” or “b.”. Since a period (“.”) occurring in this context is interpreted as a sentence delimiter by GATE’s sentence-splitter these constructs lead to erroneous decomposition of claims and are therefore removed.

In many documents the actual claim text is preceded by its claim number. Since this information is already implicitly given via the order of the claims in the patent document it is removed.

The term “characterized” is an important element that needs to be identified. The British spelling variant is replaced by the American one.

In the last preprocessing step all occurrences of the word “said” are replaced with the definite article “the”. This is a simple but effective way of improving the

**Table 10.3** Claim Types

Data Set	Claim type	Number of claims
Analyzed Set	Physical Entity Claims	114
	Method Claims	41
	Use Claims	4
Evaluation Set	Physical Entity Claims	10,310
	Method Claims	3,315
	Use Claims	3

performance of natural language parsers even before decomposing the claims. Natural language parsers trained on general language texts interpret the word “said” as a verb. In claims, however, it is always used for referring to an already introduced concept.

### 10.5.2 Claim Type and Category Identification

A simple heuristic is used to determine whether a claim is dependent or independent. The drafting guidelines for dependent claims suggest that it should consist of two parts. The first part contains a reference to the claim or claims which are refined written in a form such as “The dental handpiece of *claim 1*” or “The orthodontic bracket of any one of *claims 1 to 7*”. All claims containing either the word “[Cc]laim” or “[Cc]laims” are classified as dependent claims, all others as independent claims.

Independent claims can be categorized into: *physical entity claims*, *method claims*, *use claims*. This distinction is important, because the types differ slightly and require distinct analysis patterns. A heuristic based on keyword matching is used for this purpose. Since the developed method is based on linguistic patterns found in claims and does not deal with any legal aspects, the defined categories may differ from the categories commonly used in the patent domain.

The examination of the Analyzed Set has shown that claims containing the keyword “method” or “process” within the first 100 characters can be classified as method claims and all claims which start with the phrase “The use” are classified as use claims. Thus, simple string matching can be used.

No such simple heuristics are available for identifying physical entity claims. Physical entity claims usually start with the claimed invention rather than with claim-specific keywords. Claims that can neither be classified as use claims nor as method claims are classified as physical entity claims.

Table 10.3 shows the frequency of each claim category in the two data sets. The figures show that the number of physical entity claims is about three times higher than the number of method claims and it can also be seen that almost no use claims are present in the data sets.

### ***10.5.3 Claim Decomposition***

Our process of decomposing claims consists of three main phases: pattern identification, pattern extraction, post processing and merging the extracted parts into a tree structure. Some patterns can be identified through simple lexical matching of keywords. If this is possible, patterns are identified using Java regular expressions. Most patterns, however, are more complex and thus require deeper linguistic analysis of the claim. Therefore, the claims are analyzed with GATE<sup>5</sup> an open source natural language processing framework. Each claim is tokenized and a sentence-splitter is applied. Depending on the requirements of the extraction rules, Parts-Of-Speech tagging and Noun Phrase Chunking is done.

Based on the annotations created by the rules (JAPE grammars) the claims can be decomposed. For this purpose the textual content of each annotated pattern is extracted from GATE's internal flat document representation into a GATE-independent hierarchical tree data structure. For each extracted part a number of post processing steps are executed to remove unnecessary characters such as white spaces, punctuation symbols and words from the extracted parts.

The decomposed claims are stored in a tree structure. Each node in the tree contains an extracted part of the claim. The edges represent the relation type to the parent. Each node contains the text of the extracted part and, to be able to traverse the tree, a reference to its parent relation and a list of child relations. Each relation contains an enumerated type indicating the type of the relation and an optional string containing a label for the relation.

### ***10.5.4 Independent Claim Decomposition***

Due to space considerations, we focus more on the decomposition of independent claims in this article, since they are longer and more complex than dependent claims and thus more interesting. Due to large structural differences of claims from different categories only a very limited number of rules which are applicable to all claim types is available. The major part of the developed rules is specific to one of the claim categories. In the following section the extraction rules for physical entity claim are described.

#### **10.5.4.1 General Patterns**

Before a claim is decomposed using the claim category-specific rules the following two patterns are extracted.

---

<sup>5</sup><http://gate.ac.uk/>.

**Claim-Subject** A claim-subject is extracted and used as the root node of the tree structure. The claim-subject is that part of the claim to which all other claim parts are directly or indirectly related to. For method and use claims the identification of the subject is rather trivial. In method claims all other extracted parts can be attached to the initial keyphrase “A method” or “A process”. For use claims they can be attached to the phrase “The use”. While the claim-subject for these two categories can be extracted using a simple string matching approach, this is usually not the case for physical entity claims. In physical entity claims the root of the sentence is the invention itself. This is illustrated in Example 1. Therefore each claim sentence is analyzed with GATE and the first noun phrase is extracted as claim-subject.

*Example 1* (EP1444966-A1)

Claim-Subject

A dental head unit capable of measuring a root canal length of a patient

**Characterized-Pattern** If a claim is drafted in the two-part form as suggested by the EPO, the keyphrases “characterized in that” and “characterized by” can be used to split the claim into the preamble and the claim body. This pattern can be exploited without linguistic analysis. Regular expressions are used to split the claim text where either of the keyphrases mentioned above occurs. The characterized-part (claim body) is attached to the root of the tree structure with a CHARACTERIZED relation. For physical entity claims the characterized-part is further analyzed with the rules described in Subsection “Characterized-Part Decomposition” of Sect. 10.5.4.2. The preamble itself is not attached to the tree structure. It is decomposed using the category-specific rules described in the following sections. If a claim does not contain a Characterized-Pattern, the entire claim text is decomposed using these claim category-specific rules.

### 10.5.4.2 Physical Entity Claims

The focus in this method was set on the analysis of physical entity claims. Due to the comparatively large number of physical entity claims in the Analyzed Set, it was possible to identify a larger number of patterns.

**Composition-Pattern** The pattern which occurs most frequently in physical entity claims is the Composition-Pattern since an invention is usually described by enumerating all elements it is composed of. Thus the complexity of claims can be significantly reduced by correctly extracting these elements. The Composition-Pattern is introduced by one of the keywords “comprising”, “comprises” or “including” and is composed of several composition-parts. Each of these composition-parts describes an element of the invention and therefore starts with the introduction of a new concept. The parts can be identified by looking for singular or plural noun phrases preceded by the indefinite article “a” or “an” such as shown in Example 2.

*Example 2* (EP0063891-B2)

$$\overbrace{\text{An ejector holder [...] , the holder comprising}}^{\text{Claim-Subject}}$$

$$\underbrace{\hspace{10em}}_{\text{Composition-Start}}$$

$$\overbrace{\text{an elongate barrel [...] ,}}^{\text{Composition-Part}} \underbrace{\text{a plunger [...] , [...] }}_{\text{Composition-Part}} \overbrace{\text{and a lever [...] .}}^{\text{Composition-Part}}$$

$$\underbrace{\hspace{10em}}_{\text{Composition-Part}}$$

The JAPE grammar used for extracting Composition-Patterns first annotates the start of a Composition-Pattern by looking for the keywords mentioned above. It then identifies and annotates the composition-parts the Composition-Pattern is composed of by looking for singular or plural noun phrases preceded by the indefinite article “a” or “an”. The grammar takes into account several different linguistic patterns in order to identify nested Composition-Patterns. Each extracted part is attached to the CLAIM-SUBJECT with a COMPOSITION relation.

**Nested-Sentence-Pattern** Since each claim has to be written in one sentence, certain grammatical structures are used for chaining separate sentences to create one single sentence. A very common structure used for this purpose is the Nested-Sentence-Pattern where an already introduced concept is refined. Example 3 shows the typical structure of a Nested-Sentence-Pattern.

*Example 3* (EP0028529-B2)

$$\overbrace{\text{A scaler tip}}^{\text{Claim-Subject}} \underbrace{\text{having an operative end}}_{\text{Description}}$$

$$\underbrace{\hspace{10em}}_{\text{Nested-Sentence-Part}}$$

$$\text{, the operative end terminating in a curved free end}$$

$$\underbrace{\hspace{10em}}_{\text{Nested-Sentence-Part}}$$

$$\text{, the operative end having a non-abrasive working portion}$$

There are several, very similar, keyphrases which introduce a nested sentence. The phrases “, the CONCEPT” or “; the CONCEPT” where CONCEPT represents an already introduced concept are used frequently. In the original claims the word “said” is often used instead of the article “the”. However, since all occurrences of the term “said” are replaced by the word “the” during the preprocessing steps only the keyword “the” has to be taken into account. A nested sentence ends when either another Nested-Sentence-Pattern is found or the sentence ends. The extracted sentences are attached to the claim-subject node with a NESTED-SENTENCE relation.

**Description-Pattern** All words between the claim-subject and the first pattern found in the claim (Nested-Sentence or Composition-Pattern), are extracted as description-part. The description usually indicates the purpose of the invention (see Example 4). In some cases, however, it describes elements that an invention contains.

*Example 4* (EP0415508-A2)

Claim-Subject

An apparatus to continuously harden light curing resins, comprising [...]

Description

A JAPE grammar is used to annotate all words after the claim-subject until either a Nested-Sentence-Pattern or a Composition-Pattern is found or the claim sentence ends. The annotated part is extracted and appended to the claim-subject node in the data structure with a DESCRIPTION relation.

#### 10.5.4.3 Characterized-Part Decomposition

If the claim is drafted in the two-part form as suggested by the EPO, the characterized-part extracted with the Characterized-Pattern rule can be decomposed further into smaller parts. The annotation and extraction process first looks for extractable enumerations of elements. To this end, the Composition-Pattern rules are used in a slightly modified version. The extracted parts are attached to the node containing the characterized-part with a COMPOSITION relation.

Parts of an invention specified in the characterized-part are not necessarily enumerated using a Composition-Pattern. In some cases the parts are simply separated by semicolons. Therefore, if no Composition-Pattern is found, the characterized-part is simply split by semicolons. If this results in more than one part, each of these parts is added to the node containing the characterized-part with a CHARACTERIZED-COMPOSITION relation.

#### 10.5.4.4 Composition-Part Decomposition

Extracted composition-parts can be further decomposed by splitting them into a part containing the element of the invention and a second part containing a description of the element. This is illustrated in Example 5.

*Example 5* (EP1484028-A2)

Element-Part                      Description-Part

[...] a chuck assembly secured to the rotor shaft

Element-Part                      Description-Part

[...] a positioning template for guiding the positioning and bonding [...]

A JAPE grammar is used to identify the end of the element-part by looking for specific linguistic patterns like verbs in gerund form possibly preceded by an adverb (“a neck section *extending proximally* from the head section [...]”) or verbs in past tense, possibly preceded by an adverb (“a brush part *detachably attached* to one end of the drive shaft”). The element-part remains in the already existing composition-part node. The extracted description is added to it with a COMPART-DESCRIPTION relation. The description-part itself can be decomposed into even smaller units by extracting nested sentences. This is done using the Nested-Sentence-Pattern rule.

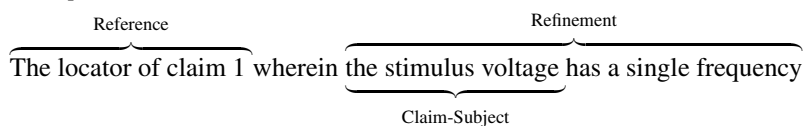
### 10.5.5 Dependent Claim Analysis and Decomposition

Dependent claims consist of two parts. The first part provides a reference to the claim(s) it refines while the second part describes the refinement itself. The analysis of dependent claims consists of two tasks. In the first step the reference-part is analyzed to extract the references to refined claims. References to previous claims are provided in various forms like as a single number, an enumeration of numbers, a range of numbers and sometimes as written text. For each of these cases several rather similar patterns have to be taken into account. The most important ones are single numbers and ranges of numbers preceded by the word claim such as in “The locator of *claim 1* wherein [...]” or “An article as claimed in any of *claims 12 to 14*, wherein [...]”. The annotated references are extracted and evaluated. Each claim object in the internal data structure is assigned a list of dependent claims based on the extracted claim reference numbers. These references can then be used to assign each dependent claim to all the claims it refines.

In the second phase the claim is split into a reference and a refinement-part. For dependent physical entity claims the refinement-part is decomposed with rules similar to those used for decomposing independent claims.

First the claim is split into two parts, the reference-part and the refinement-part. A JAPE grammar is used to identify the end of the reference-part according to several linguistic patterns. In the most commonly used pattern the reference-part ends with one of the phrases “, wherein”, “, characterized in that” or “characterized by” such as in “Hinge member as claimed in claim 1, *wherein* the head means is circular [...]”.

Then, as for independent claims, a claim-subject is extracted as the root node of the tree data structure. For this purpose the first noun chunk in the refinement-part is extracted, if it is an already introduced concept. This means that it either starts with the word “the” or “each”. Example 6 provides a better understanding of the claim-subject extraction rule. If no valid claim-subject can be found, the label of the root element of the tree structure is left empty. The refinement-part is added to the claim-subject node with a REFINEMENT relation, the reference-part with a REFERENCE relation.

*Example 6* (EP0171002-B1)

Finally the refinement-parts extracted from dependent physical entity claims are decomposed further by extracting Composition as well as Nested-Sentence-Patterns. The rules for extracting Nested-Sentence-Patterns are the same ones which are used in the decomposition of independent physical entity claims. The Composition-Patterns are extracted with the same rules used for decomposing characterized-parts from physical entity claims (see Sect. 10.5.4.2).

### 10.5.6 Merging of Dependent and Independent Claims

After the claims have been analyzed and decomposed, a coreference resolution algorithm is applied for merging each independent physical entity claim with its direct and indirect dependent claims. For this purpose the refinement-parts extracted from dependent claims are attached directly to the node in the tree data structure where the refined element was introduced. For attaching refinements from dependent claims to the correct node in the tree structure of the independent claim, the noun phrase introducing the refined element has to be found. For this purpose, the fact that a new element is usually introduced with a phrase such as “a CONCEPT” and later referred to as “the CONCEPT” can be exploited. For each claim a concept index containing *New-Concepts* and *Ref-Concepts* is created. The merging algorithm is illustrated in Example 7, showing the decomposition of an independent and a dependent claim and how the dependent claim can be merged into the tree data structure of the independent claim. The refinement-part “the base member consists essentially of [...]” from the dependent claims is directly attached to the composition-part “a base member”, introducing the refined element in the independent claim.

*Example 7* (Claims Before Merging)**Independent claim:**

An oral appliance for placing in a mouth of a user, the appliance comprising: a base member having a generally U-shaped form corresponding to the outline of a jaw of a user, [...]

```
Subject: An oral appliance
Relation: DESCRIPTION
      ->for placing in a mouth of a user
Relation: COMPOSITION
      ->a base member
      Relation: COMP_PART_DESCRIPTION
            ->having a generally U-shaped form
                corresponding
                to the outline of a jaw
                of a user [...]
```



**Dependent claim:**

An oral appliance according to any one of claims 1 to 3, wherein the base member consists essentially of a rigid plastics material which is polyethylene.

Subject: the base member  
 Relation: REFERENCE  
     ->An oral appliance according to any one  
             of claims 1 to 3  
 Relation: REFINEMENT  
     ->the base member consists essentially  
             of a rigid [...]

**Merged claims:**

Subject: An oral appliance  
 Relation: DESCRIPTION  
     ->for placing in a mouth of a user  
 Relation: COMPOSITION  
     ->a base member  
         Relation: COMP\_PART\_DESCRIPTION  
             ->having a generally U-shaped form  
                 corresponding  
                 to the outline of a jaw  
                 of a user [...]  
         Relation: REFINEMENT  
             ->the base member consists  
                 essentially of a rigid [...]

**Reattachment of Claim Parts** In some cases nested sentences or characterized-parts extracted from independent claims are not attached to the node where the element they refine was introduced. Thus a similar procedure as for attaching the refinement-parts extracted from dependent claims is used for reattaching these parts. The first Ref-Concept found in the nested sentence or characterized-part is used to find nodes in the tree structure where the parts may be attached to. For this purpose a similarity measure is computed for the selected Ref-Concept and each New-Concept in the concept index of the independent claim. The part is reattached to the node with the best matching New-Concept provided that the Levenshtein similarity value for the two concepts is larger than 0.7. Otherwise the part remains attached to its original parent.

## 10.6 Evaluation

### 10.6.1 Independent Claim Decomposition

In this section it is evaluated how the method developed in this work reduces the length and complexity of independent claims. To this end the average length of the original independent claims is compared with the average length of parts extracted from these claims. The coverage of the Stanford Parser is used as a measure for complexity reduction. In order to provide an estimation of the quality of the rule

**Table 10.4** Length reduction: independent claims

Data set	# Parts	Avg. claim length	Avg. part length
Analyzed Set	1,012	127.81	18.95
Evaluation Set	100,291	132.33	16.95

**Table 10.5** Length reduction comparison for claim categories

Data set	Claim category	# Parts	Avg. part length
Analyzed Set	Physical Entity claims	859	15.90
	Method and Use claims	153	36.06
Evaluation Set	Physical Entity claims	85,757	15.16
	Method and Use claims	14,534	27.54

sets 15 physical entity claims selected from 15 different patents and 10 method claims selected from 10 different patents, were manually analyzed and checked for correctness. Due to their small number in both data sets use claims were excluded from the evaluation. Since no gold standard is available, this evaluation was done by manually classifying the claims as “correct/mostly correct”, “partly correct” and “incorrect/insufficiently decomposed”.

Table 10.4 shows the number of extracted parts and the average number of words per part for the Analyzed Set and the Evaluation Set and compares them to the average claim length of the unparsed claims. The application of the extraction algorithm shows very promising results in terms of length reduction of independent claims. For the Analyzed Set the average part length is reduced by about 85% compared to the original claim length. For the Evaluation Set a reduction of about 87% is achieved. The results incorporate all extracted claim parts except the claim-subject since it normally consists of only about three words and would therefore distort the average number of words per part and the average number of successful parses.

The good performance on the Evaluation set indicates that the rules are generic enough to achieve a high reduction of complexity for all patents from the IPC category A61C. It also indicates that the decomposition algorithm cannot only be applied to European patents but can also handle the structurally slightly different US patents.

Table 10.5 compares the average length of parts extracted from physical entity claims with the average length of parts extracted from claims belonging to the other two categories for both data sets. The figures show that the average length of physical entity claim parts is less than half of the average length of method and use claim parts. This reflects the fact that the decomposition rule set for physical entity claims is much larger than the one for method claims and shows the positive results of decomposing extracted claim parts into smaller sub-parts.

The achieved complexity reduction can be estimated from the number of successful parses using the Stanford Parser. Table 10.6 shows the success rate of the

**Table 10.6** Stanford parser success rate: extracted parts

Data set	JVM max. heap size	Successful parses	Failed parses	% of successful parses	Improvement
Analyzed Set	1000 MB	1, 010	2	99.80%	<b>16.79%</b>
	500 MB	1, 003	9	99.11%	<b>43.14%</b>
Evaluation Set	1000 MB	100, 140	151	99.85%	<b>21.55%</b>
	500 MB	99, 793	498	99.50%	<b>44.60%</b>

**Table 10.7** Quality estimation: physical entity claims

	Count	Percentage
Correct	9	60.00%
Partially correct	2	13.33%
Incorrect	4	26.67%

**Table 10.8** Quality estimation: method claims

	Count	Percentage
Correct	4	40.00%
Partially correct	2	20.00%
Incorrect	4	40.00%

parser applied to the parts extracted from the Analyzed Set and the Evaluation Set with the same JVM heap size settings used for parsing the original non-decomposed claims. The last column shows the improvement compared to applying the parser to the original claims. The comparison shows that the coverage of the Stanford Parser is significantly higher on the extracted parts than on the original claims with the improvement being even slightly higher on the Evaluation Set.

The overall quality estimation of the decomposition rules for physical entity claims is very promising in terms of accuracy and coverage. Most of the evaluated claims are either decomposed correctly or with minor errors. Only very few claims were found which are classified as physical entity claims but are structurally too different to be handled properly by the rules. The evaluation results are shown in Table 10.7. From the 15 analyzed claims nine are decomposed correctly or almost correctly, two are considered partially correct and four are classified as incorrect or insufficiently decomposed.

Table 10.8 shows the evaluation results for the 10 analyzed method claims. The figures show that four claims are decomposed correctly, two are partially correct and four are insufficiently or incorrectly decomposed. The detailed evaluation shows that the performance of the developed decomposition rules varies greatly depending on the structure of the claims. Method claims which consist of an enumeration of steps, wherein each step starts with a verb in gerund form, are decomposed correctly. Some claims on the other hand also provide a description of materials or apparatuses used

**Table 10.9** Resolved claim references

		Total Number	Percentage
Analyzed Set	Attached claim references	81	96.43%
	Missing claim references	3	3.57%
	Total number of dependent claims	84	100%
Evaluation Set	Attached claim references	77	100%
	Missing claim references	0	0%
	Total number of dependent claims	77	100%

for carrying out the method or enumerate steps in a form that cannot be handled correctly by the rules.

### 10.6.2 Claim Merging

From each of the data sets, 10 patents containing a physical entity claim were randomly selected and evaluated manually in terms of correct attachments, incorrect attachments and the number of parts for which no attachment was found. For the parts which could not be attached, it is differentiated between parts for which no claim-subject was found and those part which could not be attached although a claim-subject was identified by the rules. For the dependent claims, for which no subject could be found, it is analyzed whether the claim-subject does not exist or it was not identified by the decomposition rules.

Table 10.9 shows the performance of the rules used for resolving references from dependent claims. The row “Attached claim references” shows for how many dependent claims the reference to their parent was correctly resolved while the row “Missing claim references” shows how many claims could not be attached to the claim they refine. The figures show that for all independent claims selected from the Evaluation set the dependent claims were attached successfully. In the Analyzed Set the claim reference was not successfully extracted for two dependent claims.

Table 10.10 provides an overview of the performance of the claim merging process for the Analyzed Set and the Evaluation Set. The row “Correct attachments” shows how many parts were attached correctly to the part they refine and the row “Incorrect attachments” shows how many parts were attached erroneously.

In the row “No claim-subject/correct” it can be seen how many dependent claims did not have an extractable claim-subject. The row “No claim-subject/incorrect” shows for how many dependent claims a claim-subject existed but was not found

**Table 10.10** Attachments

		Total Number	Percentage
Analyzed Set	Correct attachments	33	40.74%
	Incorrect attachments	5	6.17%
	No attachment found	24	29.63%
	No claim-subject/correct	9	11.11%
	No claim-subject/incorrect	10	12.35%
	Attached claim references	81	100%
Evaluation Set	Correct attachments	36	46.75%
	Incorrect attachments	1	1.30%
	No attachment found	32	41.56%
	No claim-subject/correct	2	2.60%
	No claim-subject/incorrect	6	7.79%
	Attached claim references	77	100%

by the rules. The figures show that the number of correct attachments is relatively high while there are almost no incorrect attachments. The figures also show that the percentage of parts for which no attachment was found is relatively high in both data sets. One reason is that a *Ref-Concept* in a dependent claim can be provided in a shorter form than the original *New-Concept* as for example a concept may be introduced as “spaced-apart arms” in an independent claim and referenced with “the arms” in the dependent claim.

Another reason is that some dependent-claim-subjects are not extracted correctly due to erroneous POS-tagging. This affects especially the term “means”. This occurs for phrases such as “The impression tray according to claim 1 in which the *light-reflecting* means comprises a thin layer of reflective metal.”. In this case the term “the light-reflecting” is extracted as the claim-subject instead of the term “the light-reflecting means”. A possible solution would be to create a specific rule for the term “means” in a similar way as is followed for extracting composition-parts.

The third reason is that the extracted claim-subject is not always the concept which is refined. This is shown in the phrase “The impression tray according to claim 5 in which *the edges of the cover sheet* are sealed to [...]” where the term “the edges” is extracted as claim-subject instead of the words “the cover sheet”.

This problem is also reflected in the number of dependent claims for which erroneously no claim-subject was found. Most of those claims follow a structure where the concept to which the part should be attached is written at the end of the sentence such as in the claim “A teeth straightening bracket according to claim 1 characterized in that engaging fingers [...] are disposed except for the both longitudinal ends of *the wire support*”.

## 10.7 Conclusions and Future Work

We have shown that the automatic analysis of patent claims using natural language parsers can be dramatically improved by decomposing them first into smaller units using a set of rules and heuristics. This research is a first step toward developing sophisticated methods and tools to facilitate the work of patent information professionals by automatically analyzing, structuring and visualizing patent claims.

The developed method shows that rule-based decomposition of patent claims is feasible due to the particular language used for drafting patents. The evaluation shows promising results in terms of reduction of length and complexity of independent claims and shows that the decomposition method eases the application and raises the performance of existing information retrieval and information extraction tools. A quality estimation for the correctness of the extracted parts shows good results for physical entity claims where a high percentage of evaluated claims is decomposed either correctly or with minor errors. While the decomposition rules seem to be detailed enough for physical entity claims, additional work has to be done for method claims as the extracted parts remain very often long and complex. Further analysis has also to be done for dependent method claims for which currently no decomposition rules exist. The procedure for merging dependent and independent claims has to be extended and adapted for method claims. Particularities of dependent method claims will have to be taken into account, as refinements may be provided in different forms than in dependent physical entity claims. Regarding the claim merging procedure for physical entity claims it should be evaluated how the quality of the results changes when different string similarity measures and thresholds are used. It should also be evaluated how the results change when other terms are used for attaching the claim when no attachment can be found for the dependent-claim-subject.

The evaluation on a large data set has shown that the rules created from the analysis of a small data set containing only European patents are generic enough for the IPC category A61C and that they can also be applied to US patents. Since the rule set does not use any domain-specific keywords it is very likely that the rules can also be applied to patents from other IPC categories. To test this hypothesis further evaluation needs to be done on a data set containing patents from a wider range of IPC categories in order to see how the performance of the rules depends on the domain of the invention.

An important aspect regarding evaluation is to seek intensive cooperation with researchers from the intellectual property domain for developing gold standards and precise criteria for measuring the quality and the correctness of the extracted claim parts.

To our best knowledge this work is the first approach of decomposing English-language patent claims and can therefore be seen as a starting point for additional work in various fields of patent information retrieval. Besides the visualization of decomposed claims for improving readability as done in this work, the method can be used for tasks such as document retrieval or computing structure-based similarity measures. It can therefore be a contribution to the development of information

retrieval methods especially tailored to the patent domain needed by various parties such as patent offices, patent attorneys and inventors.

## References

1. Fellbaum C (ed) (1998) WordNet: An electronic lexical database (language, speech, and communication). MIT Press, Cambridge
2. Guidelines for examination in the European Patent Office. <http://www.epo.org/patents/law/legal-texts/guidelines.html>, last visited: 2009-12-08. European Patent Office, Status April 2009
3. Indukuri KV, Ambekar AAA, Sureka A (2007) Similarity analysis of patent claims using natural language processing techniques. In: Proceedings of the international conference on computational intelligence and multimedia applications (ICCIMA'07), Sivakasi, India. IEEE Computer Society, Washington, pp 169–175
4. Manual of Patent Examination Procedure (MPEP) (2008) <http://www.uspto.gov/web/offices/pac/mpep/mpep.htm>, last visited: 2009-12-08
5. Sheremetyeva S (2003) Natural language analysis of patent claims. In: Proceedings of the ACL-2003 workshop on patent corpus processing, Sapporo, Japan. Association for Computational Linguistics, Stroudsburg, pp 66–73
6. Shinmori A, Okumura M, Marukawa Y, Iwayama M (2003) Patent claim processing for readability: Structure analysis and term explanation. In: Proceedings of the ACL-2003 workshop on patent corpus processing, Sapporo, Japan. Association for Computational Linguistics, Stroudsburg, pp 56–65
7. Verberne S, D'hondt E, Oostdijk N, Koster CH (2010) Quantifying the challenges in parsing patent claims. In: Proceedings of the 1st international workshop on advances in patent information retrieval (AsPIRe 2010), Milton Keynes, UK, pp 14–21
8. Yang S-Y, Soo V-W (2008) Comparing the conceptual graphs extracted from patent claims. In: Proceedings of the 2008 IEEE international conference on sensor networks, ubiquitous, and trustworthy computing (SUTC 2008), Taichung, Taiwan. IEEE Computer Society, Washington, pp 394–399

# Chapter 11

## From Static Textual Display of Patents to Graphical Interactions

**Steffen Koch and Harald Bosch**

**Abstract** Increasingly, visualisation is becoming a crucial part of patent search and analysis tools. Due to the benefits for accessing and presenting large amounts of data quickly, it is highly relevant for common tasks in the intellectual property domain. Graphical representations become an even more powerful instrument by adding interactive methods allowing for the user-guided-change of perspectives and the exploration of the presented information. A close integration of interactive visualisation into search and analysis cycles can leverage seamless search and analysis environments, as proposed for similar tasks in the relatively new research field of visual analytics. This chapter proposes such a visual analytics approach for the intellectual property domain. One possible way to accomplish this integration is shown on the basis of the research software prototype PatViz. The chapter contains a discussion of the benefits as well as the difficulties arising through the realisation of such a system as well as an outlook on how the methods can be exploited for collaboration tasks.

Today, the amount of generated digital data is increasing rapidly and a large part of these data is textual information. While the means to search for and within text documents have matured for some areas, such as web search, the fields of text mining and high quality text analysis still pose a variety of problems. On the one hand, these are intrinsic problems of natural language processing (NLP), information extraction and information retrieval. On the other hand, human users have to formulate their information need during search tasks, they have to interpret the results of search and text analysis and they must be able to assess the quality of these results. The latter are mainly perceptual and cognitive aspects.

The above-mentioned research fields, such as information retrieval, etc., already provide (semi-)automatic methods and techniques, which relieve users of the burden to read or skim through all available textual information. However, contextual information that is required in order to judge a search result's quality or to provide the necessary background for formulating and refining a user's information need is typically provided in textual form. Reading text takes time, especially when high

---

S. Koch · H. Bosch

Institute for Interactive Systems and Visualization, Universität Stuttgart, Stuttgart, Germany



quality text analysis is of relevance. Many patent analysis tasks do require this type of high quality analysis.

From the authors' point of view, there ultimately is still no alternative to reading a patent document, when it comes to understanding all its technical and legal details, even considering the impressive progress in natural language processing and similar fields within recent years. However, interactive visualisation orchestrated in an intelligent manner presents a good opportunity for shortening analytic cycles during patent analysis, thus helping patent searchers in building trust in their queries faster, performing analytic steps more quickly, and leverage mining techniques that are not very common in patent analysis tasks, since they are difficult to understand and control without the provision of visual feedback.

To achieve these goals, a much closer integration of retrieval and NLP tools with visualisation has to be accomplished as well as their seamless embedding into the patent analysis process itself. Such integration can only be achieved by combining visualisations with mechanisms allowing for interactive connection to search and analysis services. The research area of Information Visualization (cf. [1, 2] and [3]) provides a multitude of techniques and methods to explore, present and understand abstract information. This also encompasses a broad variety of techniques for document and text analysis.

In recent years, a new research field called 'visual analytics' has been established. An overall goal is to address situations where users have to deal with huge amounts of data and where fully automated solutions are not applicable, since the analysis requires human insight, judgement and the ability to make complex decisions. For certain tasks, the combination of automatic analysis techniques and visualisation techniques can help to bridge this analytical gap by including human analysts directly in such a process. Interaction plays a vital role in making this combination beneficial to analysts, since it creates the glue needed for steering the exploration and analysis of its outcome. Hence, visual analytics encourages a more holistic view of the problem space or task at hand. As mentioned above, general scenarios that profit from visual analytics approaches are comparable with the situation for patent analysis tasks.

This chapter highlights the fundamental aims of visual analytics and argues how they can be applied to and exploited for patent analysis. This encompasses the discussion of a variety of aspects, such as patent visualisation and interaction techniques, users and tasks, integration into a patent analysis processes—logically as well as from a technical point of view—and an outlook on different devices and platforms suitable for "visual patent analysis". In order to provide an overview of current and future research for possible solutions and corresponding design requirements, one particular example from a current research prototype is included.

## 11.1 Visual Patent Analysis

In 2005, Thomas and Cook published 'The Research and Development Agenda for Visual Analytics' [4]. Even though their focus is on the prevention of terror-

ist threats, recovery from natural disasters and other emergency situations, the general approach they propose for visual analytics can be adopted to a large variety of civil application and business use cases, as Keim et al. describe in ‘Mastering the information age—solving problems with visual analytics’ [5]. Accordingly, many works in visual analytics research deal with document analysis in a broader sense, because intelligence data often come in written form, and textual business news are important in financial analysis tasks, for example. This trend is clearly reflected in those research papers published on the ‘IEEE Symposium on Visual Analytics Systems and Technologies’,<sup>1</sup> which apply the approach to fraud detection in the finance sector [6] or investigative analysis [7].

As all visualisation research disciplines, visual analytics exploits the exceptional characteristics of human visual perception. More information is perceived through this channel than through all other human senses taken together. Due to the broad bandwidth of the human optical apparatus, visual information as provided in form of images can be processed in a highly parallel manner. This holds true for special visual properties such as shapes and edges. Unfortunately, the situation is different when it comes to reading written text. Evidence suggests that additional parts of the human brain are involved here, in comparison to those responsible for visual interpretation. As a consequence, parallel perception, at least of considerably large amounts, of contiguous textual information is hardly possible [8].

However, spoken and written natural language is the most ubiquitous information and natural language supplies us with the world’s most developed symbol system. Providers of patent analysis tools must therefore decide carefully for which task and how visualisation can be successfully exploited for patent analysts. As mentioned above, images can support users in perceiving rather large amounts of information in parallel, but they are not a suitable substitution for textual information. In cases where the understanding of facts can be transported more quickly or where additional insight can be drawn from visualisation, its application is justifiable and useful. Nevertheless, textual information and images are often used in combination, for example, in form of labels, since they are important, particularly in a textual domain, such as IP, to provide users with the necessary context, particularly in a textual domain, such as intellectual property. Interactive visual means can bridge the gap between obtaining a high level overview and detailed information, at least to certain degree. The closer one moves towards a very detailed view, ending at the textual content itself, the more difficult it becomes to aggregate and summarize the textual meaning visually. However, even on levels very close to textual representation, visual hints, such as highlighting named entities and semantic relations, can help users to gain insights into the reasons why a specific document made it into the result set. Subsequently these hints can be used to restrict or widen a query for further iterations if it does not meet a user’s expectation. Techniques for visualising textual information in digital libraries and knowledge domains are described in Börner et al. [9, 10].

---

<sup>1</sup>In 2010, the former ‘IEEE Symposium for Visual Analytics System and Techniques’ is a full conference collocated with the ‘IEEE Conference on Visualization’ and the ‘IEEE Conference on Information Visualization’, all three being part of the enclosing VisWeek event.

The key for integrating visualisation into analytic processes is interaction. Interaction enables users to drill down on interesting aspects identified in a visual representation, for example by providing zooming, panning and filtering, in order to access the details they are looking for. This very common approach has been aptly expressed by Shneiderman's [11] information seeking mantra: 'Overview first, zoom and filter, then details on demand'.

The research discipline of information visualisation is concerned with finding meaningful interactive visual representations for abstract information. Closely related is the field of human-computer interaction (HCI), where the research effort focuses on cognition and perception as well as on how users can be supported in their tasks with computers. Hereby, hardware and software issues are covered. Hence, interaction with visualisation and graphical user interfaces is, of course, a part of this wider area of interaction research.<sup>2</sup>

Visual analytics is a research discipline that spans the fields of visualisation, human-computer interaction, data mining and machine learning (including clustering and information retrieval, which have been discussed in previous chapters of this book). This is completed with the investigation of suitable models capable of dealing with large amounts of potentially heterogeneous, ambiguous and uncertain data. Visual analytics can therefore be regarded as a branch of scientific research that deals with challenges arising from the vast amounts of digital information produced by today's society. Concepts developed in this field therefore have the potential to be exploited successfully in patent analysis.

### ***11.1.1 Supporting Analytic Tasks by Exploiting Interactive Visualisation***

There exists a variety of commercially available software systems for patent search and analysis. Since these have been presented in other works,<sup>3</sup> this section neither intends to provide a comprehensive overview of current patent software nor an evaluation of them with respect to interactive visual capabilities and their integration into analytic cycles.

Instead, this section takes an academic perspective on this topic. It attempts to provide an outlook on how interactive visualisation could be exploited in order to support users in fulfilling their tasks. It also emphasises that most patent-related

---

<sup>2</sup>Very well-known are conferences such as the ACM SIGCHI. Additionally, a variety of scientific journals, for example, the ACM Transactions on Computer-Human Interaction, constitute further important resources for this area of research. The mentioned textbook [2] focuses especially on interacting with visually displayed information, while [12] provides a broader view on interaction techniques.

<sup>3</sup>Trippe describes a set of common tasks for patent search and analysis in [13], and provides an overview of commercial tools to tackle them [14]. A more recent survey can be found in Yang et al. [15]. Moehrle et al. [16] also contribute a current outline of commercial systems and relate them to a taxonomy based on a business process model.

tasks are performed in an iterative manner. Searches typically do not end after formulating the first query. Rather the query will be modified several times after examining the characteristics of the corresponding result set until the patent searcher is confident that the remaining set fulfils the requirements regarding recall and precision. The same is true for more abstract tasks, such as developing patent strategies or analysing those of competitors: hypotheses will be developed and checked against information retrieved from patent repositories. According to the findings, hypotheses will be modified and (in-)validated; again the initial query formulation may be subject to modification. These working patterns are comparable to tasks that are performed by analysts in intelligence [17]. A separation of retrieval and analysis stage, as well as its reflection in the user interface, can be counterproductive. Therefore, the direct support of these iterative patterns in an integrated and extensible manner, utilising interactive visualisation within each of them, seems a promising approach for the patent domain.

Most of the currently available tools for patent search and analysis exploit visualisation for the generation of reports. This is a valuable means for presenting analysis results, also to non-specialists, as well as for building a common understanding within collaborative tasks. Some of the tools go further by providing users with the means for interactive exploration of citation graphs, patent families and other ‘patent network’ information, typically using node-link representations or matrix-like views. Others encompass the visualisation of clustering results, aggregation of bibliographical data from patent sets, etc. In terms of supporting the overall analysis process, most of the currently available systems offer support for the subsequent tasks of search, result set presentation and access to the patent details. In general, an increase in using information visualisation techniques within patent analysis and search tools can be observed in recent years. While all of these usages are valid on their own, there is, however, still a lack of integrating them into larger analytic cycles that do not only address a very specific situation, but offer the possibilities to transport insights<sup>4</sup> gained from multiple visual perspectives and reuse them directly in subsequent tasks.

In order to exemplify usage of visualisation during patent analysis as well as its seamless integration into analytic processes, a rather abstract view on the different stages of patent analysis is provided in the following paragraph. Based on this abstract view, possible solutions demonstrating visual support for analytic loops are presented in form of the patent analysis interface PatViz<sup>5</sup> [19]. PatViz has been developed as a research prototype during the European project PatExpert<sup>6</sup> [20] in context of the framework program 6. Since 2008 the German Science Foundation

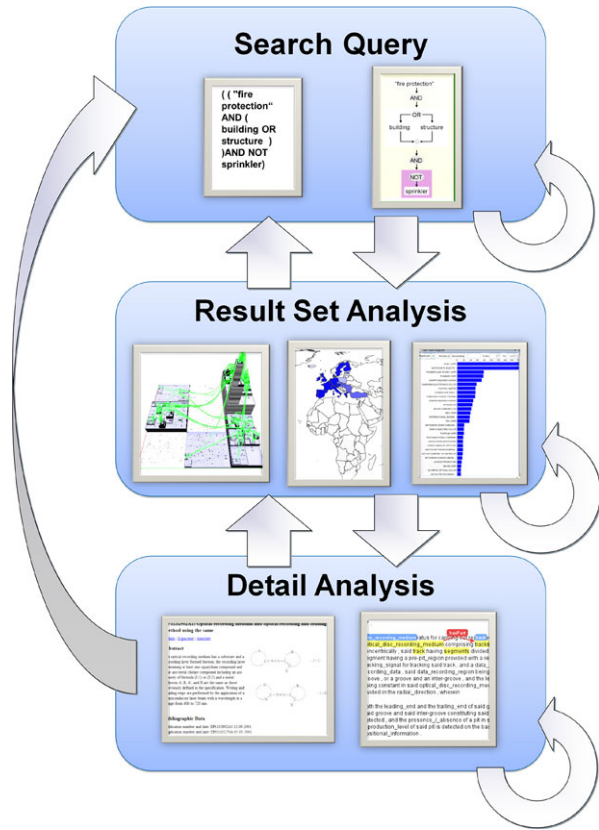
---

<sup>4</sup>Chen et al. suggest a theoretical framework for the management [18] of (visual) insights that could be used as the basis for insight transport.

<sup>5</sup>The PatViz visualization system is used here as an example for the following reasons: both authors have been involved in its development and are therefore familiar with its architecture. Furthermore, it has been developed as an academic prototype with the ideas in mind that are presented here.

<sup>6</sup><http://www.patexpert.org>.

**Fig. 11.1** The abstract patent analysis process with its three stages

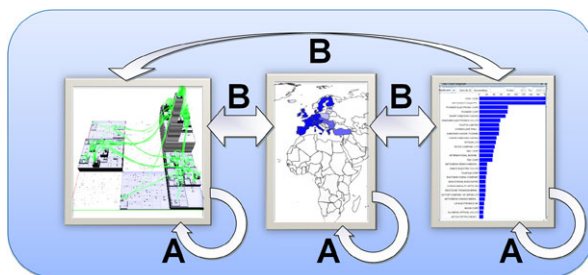


(DFG) as part of the priority programme ‘Scalable Visual Analytics’ has funded its further development.<sup>7</sup>

Many patent tasks involve searching for patents, the analysis of retrieved result sets and the detailed inspection of patent documents themselves. Typically this requires iterative adaptation of the query, driven by insights gained from intermediate patent result sets and the inspection of the documents contained therein. This abstract process is not only executed during prior art search, but similarly in cases where the analysis goals are shifted to searching for infringing patents, and in order to monitor competitors or to observe certain fields of application and technologies. Figure 11.1 illustrates the abstract process of patent analysis and the paths users may take through it. While the boxes/nodes describe the different stages, the paths depicted by arrows symbolise interactive tasks and analytic feedback loops.

<sup>7</sup><http://www.visualanalytics.de>.

**Fig. 11.2** Navigation within one stage of the abstract patent analysis process. (A) denotes interaction paths in one view. (B) indicates interaction paths across views



### 11.1.2 Visual Analysis on Single Views

At small scale such feedback loops are already useful on a single perspective (see (A) in Fig. 11.2). This holds true for all three analysis stages, including query formulation, the views aiming at result set exploration as well as the detail views. A typical example from patent analysis is the exploration and filtering of a list or table-based result set, showing the titles and other information of patents that have been retrieved according to a previously executed search request. Tools that provide such list-based interfaces usually allow for the sequential exploration of the table via scrolling and paging mechanisms, enabling users to judge how reasonable the results are according to their needs. Combined with facilities for sorting and filtering the results with regard to bibliographical data and the possibility to inspect single patents in detail, such a view is a powerful means to explore search results. It already covers a considerable part of what is stated in Shneiderman's mantra [11]. However, problems arise if the result set under analysis exceeds a certain size, since it takes time to scroll through the result list and skim through the titles, respectively, in order to judge the relevance of the results. Furthermore, if sorting is used, this allows only for inspecting the result set regarding one feature, most often not multiple features.

Solutions offering visualisation can provide different perspectives on such a result set, thereby taking advantage of the users' perceptual skills. This does not mean that tools as the one described in the previous paragraph should be replaced, but it can be of great advantage if other views offering different analytic possibilities are provided additionally. Depending on the aspect analysts are focusing on, different kinds of visualisation can be useful. If, for example, the scope of interest is countries where a certain patent or patent set is in force, presenting a map that shows these countries, plus the number of valid patents, comprises a straightforward way to transport this insight to users. Offering a variety of these visual perspectives in parallel increases the possibilities for further analysis as will be discussed in the following section. Linking different perspectives, such as the selection of a patent sub set in one of the views and its reflection in the others, increases analytic possibilities further, making them more powerful than the sum of the single views in terms of analytic possibilities. Cross verification of separate features in the patent set can help to obtain a quick overview without posing the need to investigate the presented information in a sequential manner over and over again. To some extent, such schemes can already be provided by single visualisations, such as bar charts,

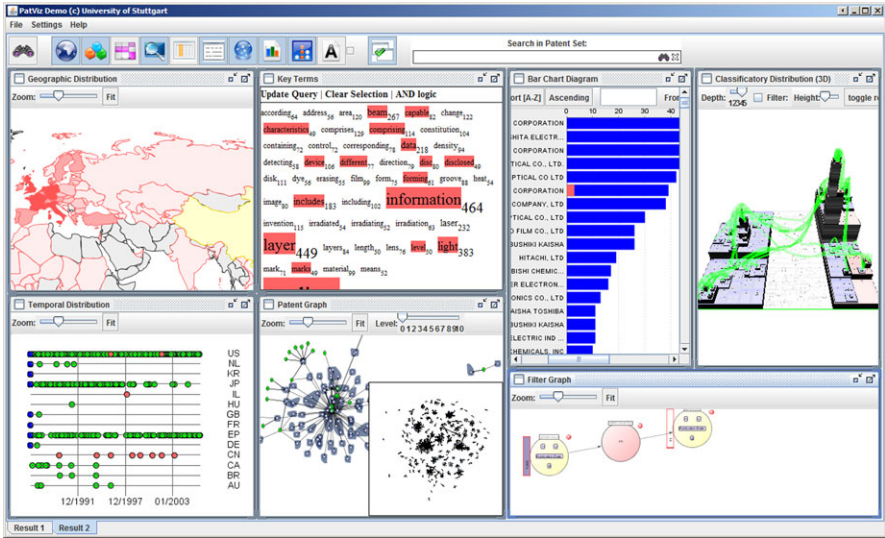
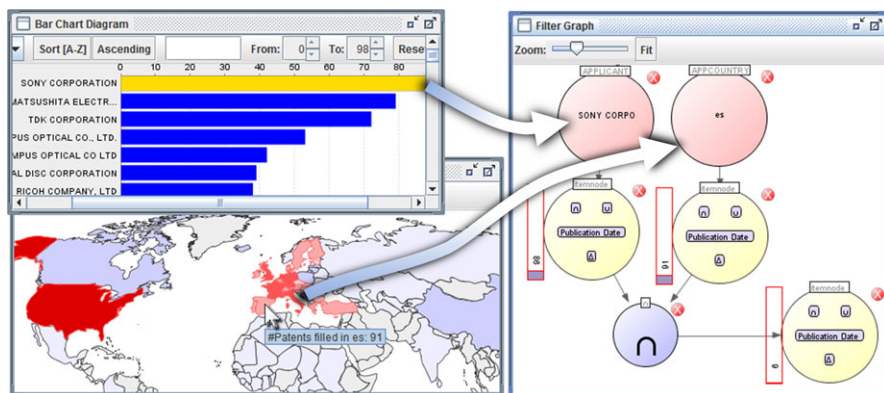


Fig. 11.3 Multiple coordinated views of a patent search result in PatViz. The currently selected aspect (P.R. China) is marked. Related aspects of the underlying patent document model are highlighted accordingly

scatter plots and parallel coordinates [21] capable of displaying two or more aspects within a single view. However, the adequacy of a visualisation technique depends on the type of data to be shown as well as on the task to be accomplished with its help. Accordingly, not all graphical perspectives are well suited to display arbitrary types of information and different methods have to be employed for showing geo-spatial, temporal, hierarchical, network-based, etc. data. A well-balanced task and data aligned selection of visual tools is therefore indispensable [11].

### 11.1.3 Using Multiple Interactive Views

Multiple coordinated views are an appropriate technique to support larger analytic cycles by integrating different perspectives of the same problem space. This corresponds to the analytic cycle depicted in Fig. 11.2. An overview of multiple coordinated views can be found in Roberts [22]. Figure 11.3 shows such an example from the research prototype PatViz. Here, users are enabled to activate and use a set of different perspectives on the same patent result set. Each of the views allows for a different set of interactions, which are adapted to the information presented, still trying to support uniform interaction gestures such as zooming and panning/scrolling, in order to lower the burden of learning how to use the views in an interactive manner. Additional mechanisms, e.g., brushing and linking, create the glue to draw further benefit from inspecting several views on the same workspace. Hereby, the selection of an aspect in one view is immediately reflected in the others through highlighting



**Fig. 11.4** The combination of aspects from different views in the selection management component. The *top nodes* in the graph depict the selected aspects. Their respective patent sets are represented by the *nodes in the middle*. The node labelled with the symbol for disjunction (*bottom left*) depicts an intersection operation resulting in another patent document set (*bottom right*)

(see Fig. 11.3). Accordingly, it is obvious to see how the selected patent documents are distributed considering other patent-related aspects. However, a variety of problems arise from introducing brushing and linking for such scenarios, aside from the issue of uniform interaction access. One such problem is how to define suitable semantics for selection in each of the views.

In a simple bar chart as shown in Fig. 11.4, which displays the applicants of a patent set under inspection and the number of applications, it is already difficult to agree on such semantics and to decide on an adequate interaction mechanism like pressing a key, clicking a mouse button or applying a mouse gesture. Given that a mouse click is used to perform selection operations on the applicants bar chart: what does it mean if a user clicks on a bar indicating that applicant X holds seven patent application within the investigated patent set? Does it reflect the selection of applicant X, or the selection of the patent documents which fulfil the property of having X as an applicant? Another interpretation could also be the selection of all applicants having exactly/more than/less than seven patent applications or again the respective patent documents. While typically some of the alternatives seem to be intuitively more reasonable than others, this example shows clearly that selection semantics have to be defined, not only for the discussed view, but also for all of them. Furthermore, this decision also influences the model that has to be created for dealing with patent documents and their properties within the visualisation and interaction module.

In PatViz the patent document was chosen as the primary object of interaction for consistency reasons regarding user interaction, as well as for building the model. This means that all selections translate to constraining properties of the current patent document set. In the case of the applicant bar chart, clicking on a node therefore results in the selection of all patent documents with the respective applicant. Nevertheless, operations such as the selection of patent documents by applicants



with a specific amount of applications in the set can be realised by sorting applicants according to their number of patents within the set and by enabling users to select several bars at once. In PatViz this kind of multi selection, however, is restricted to take place within one of the views in the multiple coordinated views and is interpreted as a union of constraints by default. The very same model is used for all the different views available for patent set inspection, which are shown in Fig. 11.3. These comprise perspectives for analysing result sets according to their distribution in classification systems, including facilities for examination of patent co-classification, the already mentioned map indicating where patents of current set are in force, a timeline view, a graph view showing document relations according to user-selectable patent properties, a tag cloud highlighting the most frequent terms in the set under inspection, and others.

In situations where different selections from multiple views should be combined in order to explore the filtering and widening effects on the patent set under analysis, the proposed approach is not feasible anymore. One possibility to allow for multi selection from multiple views is to define implicit inter-perspective Boolean operations, for example, combining the selections from different views by set operations. However, additional mechanisms are needed to support arbitrary and user-defined combinations of such selections. In PatViz this additional feature comes in form of a graph-based selection management tool. A selection can be easily transferred to this tool by drag and drop operations or by using the selection management's context menu. Besides facilities for combining selection by Boolean operations, the tool also enables users to filter according to properties of the patent subsets, to explore different combinations representing users' hypotheses in parallel, and to highlight arbitrary subsets within the graph in all views available for result set exploration. When selections are transferred to the tool, not only the set of selected patents is handled, but also the selection semantics is stored and preserved. This is an important aspect when it comes to iterative query improvement which will be discussed later in this chapter.

The tool for selection management again increases the analytical possibilities for patent analysis to an extent which exceeds the expressiveness of most available patent tools providing visual analysis. The techniques described in this section relate to the arrows depicted in Fig. 11.4. A similar technique for the exploration and filtering of multi-dimensional data sets has been proposed in [23]. Approaches using different visual representations are described in [24, 25].

With Polaris,<sup>8</sup> Stolte et al. [26] proposed a system that is not restricted to a single object of analysis and provides more analytical freedom with respect to relational analysis. It thereby combines visual information exploration, interactive analysis features and visual query mechanism intelligently. However, this approach cannot be applied in a straightforward manner any more, if different retrieval systems that do not follow a relational paradigm—for example, text retrieval systems or semantic repositories—have to be addressed by a visual frontend.

---

<sup>8</sup>The technology described in Polaris has been successfully commercialised by Tableau Software: <http://www.tableausoftware.com/>.

### ***11.1.4 Visual Support for Multiple Retrieval Facilities***

Most available tools for patent search<sup>9</sup> provide query facilities either in a form-based way or offer formal textual query languages. Normally, in both cases Boolean operators can be used for specifying and combining multiple constraints on textual content and bibliographic information. Typically, form-based approaches realise the Boolean definitions explicitly within fields, for example, by letting users search for a certain combination of keywords within a specified part of a patent document, but also implicitly by combining the fields within a form with either AND or OR operations. The forms reflect the design of patent documents, thereby providing additional hints to users regarding information they are requesting. Textual query languages normally provide a greater degree of freedom with respect to the way the single building blocks of a query can be defined. This freedom, however, comes at the cost of implicit clues about which parts of the document are addressed by a query, since there is no order of the statements required that reflects the layout of patent documents. Nevertheless, formalised query languages are the first choice for experienced users: having learned the query syntax and given the knowledge how to use them efficiently, queries can be stated in a much more straightforward fashion, in addition to the fact that they are more powerful.

During the development of PatViz several different search back-ends have been created by partners within the PatExpert consortium. These had to be supported by the visual front-end and they encompass: keyword search (text retrieval system), metadata search (bibliographic data and metadata accessible within a relational database), image similarity search and semantic search.

In order to integrate these different retrieval facilities, the Boolean paradigm for integrating them into one single query interface has been chosen in PatViz. Since Boolean search facilities are very common in patent search, combining multiple back-ends using such a strategy was a feasible solution. However, a problem arises from the fact that such search systems are addressed very differently in terms of the query mechanisms and languages they offer. As a consequence, different query languages have to be combined and, even worse, all of them have to be learned by analysts if they want to exploit the full expressiveness of the resulting system. In order to diminish these negative effects, a visual representation for building the query has been proposed as part of the PatViz prototype, allowing for both textual as well as visual query creation. The latter combines visual metaphors and textual artefacts and presents them in a way similar to the well-known ‘syntax diagrams’ [28]. Boolean AND operations are thereby represented as sequences, while OR operations are mapped to branches in the visual view. A comparable approach to the one described here has been suggested by [29]. Again, a multiple coordinated view on the two query representations has been established as is shown in Fig. 11.5. At

---

<sup>9</sup>Marti Hearst’s book, ‘Search user interfaces’ [27] gives an elaborate overview of strategies and current state of the art in searching within digital repositories. Especially Chaps. 3 and 10 are highly relevant in the context of this section.

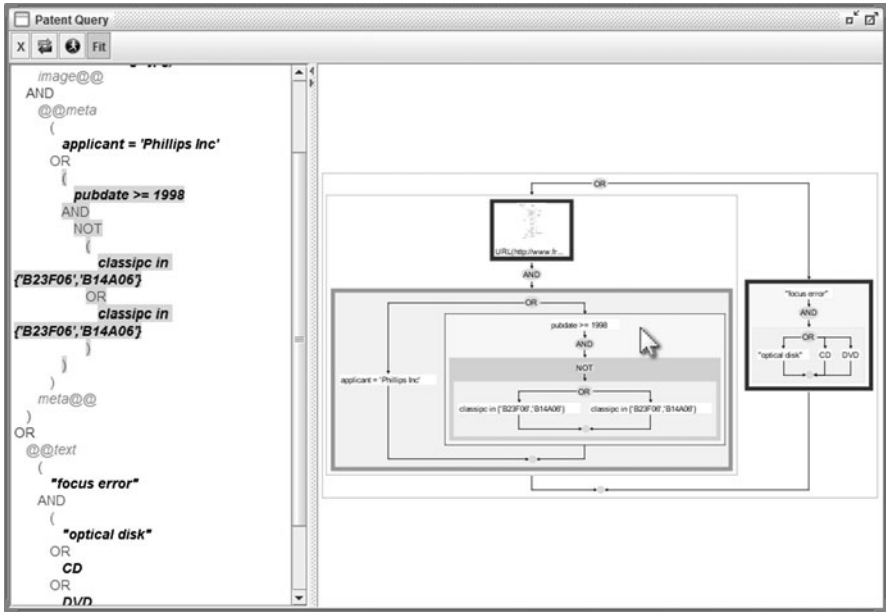


Fig. 11.5 Linked textual and visual query representations. The textual representation highlights the block on which the mouse is situated in the visual representation with a grey background

first glance there might be no obvious difference between creating a complex textual interface and representing the same complexity with visual metaphors, which, of course, also have to be learned. However, building queries visually has a number of benefits, which can help users in learning how to set up complex queries in the described environment. One advantage is the provision of a better overview and higher clarity of the single parts of the query and its overall construction by representing Boolean operators graphically. Especially large and complex Boolean constructions can be understood more easily, since operator scopes are much more apparent. A visual query representation is already useful when addressing just a single search facility. Another benefit is that only valid queries can be built by interactive visual construction. Constraints that are possibly unknown to first-time users can thereby be chosen based on the options provided in context menus. Furthermore, the visual query tool provides features for scope highlighting in both query representations and the possibility to move or delete whole query blocks by direct interaction if they are found to be placed wrongly within the complete query. Additionally, the coordinated views of the queries can be exploited for learning the textual query language, since all changes made in the visual part are reflected in the textual part.

In order to accomplish the combination of different search back-ends and their integration using the Boolean paradigm, a hierarchical query parser/generator concept has been developed. The hierarchical parser facilitates the integration of new retrieval back-ends and their corresponding query languages, simply by specifying

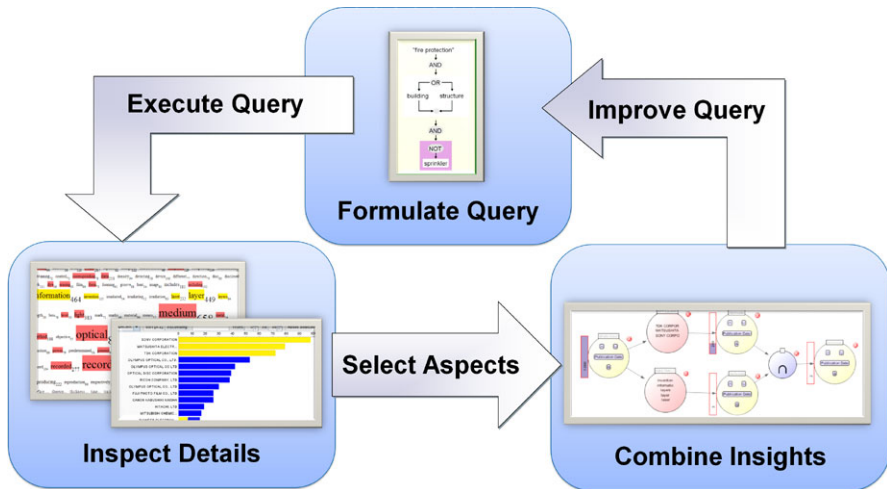
the formal grammar of the query language, resulting in the generation of the required parsing facility. The graphical metaphors, apart from representing Boolean combination, however, still have to be defined programmatically at the moment, since developing mechanisms for dynamic description of arbitrary visual representations poses a great challenge.

### *11.1.5 Closing the Loop for Iterative Patent Analysis*

Having described the analytic loops within all stages of the abstract patent analysis cycle, the loops between the different stages have not yet been addressed. Some of these inter-stage transitions are normally triggered by very common user interactions and the resulting views are displayed accordingly. This is the case for the presentation of results set views as a consequence of sending a correct query to some retrieval back-end or by loading a pre-existing patent portfolio. The same applies to showing detail views if one patent or a specific aspect of a patent is clicked, e.g., in a tabular view listing all patents of a result set. These paths are supported by all patent search and analysis tools. Unfortunately, support for reintegrating findings of result set analysis and inspection of patent details for either iterative refinement of the search or the analysis, is rare in such tools. Yet, particularly the reusing of insights gained from one step of patent analysis, including those obtained from visual representations, drive the iterative processes that are typical for patent analysis tasks. In order to accomplish interactive insight integration, the definition of appropriate selection semantics and an adequate model are important prerequisites as will be argued in the following paragraphs. Visualisation and interactive techniques can leverage this insight transport elegantly. The approach taken in the development of PatViz will again be used as an example to describe possible solutions for accomplishing this reintegration.

PatViz provides different ways to integrate important findings gained through the exploration of the patent result set perspectives and detail views. The first option is the direct selection of a specific finding from one of the views and either to drag it to the current visual query representation or to integrate it in the query using the available context menus. While the mentioned mouse gestures and visual requirements for the interaction methods can be implemented in a straightforward manner, it is not possible to guess automatically a user's underlying intention, e.g., which aspect should be constrained or widened. To some extent this problem can be resolved by applying the selection semantics in the same manner as discussed in the paragraph on the usage of multiple views. Thereby, the defined semantics implicitly determine the constraints that are transported into the query to be extended. An example is the selection of a term from a tag cloud view dragged into the visual query representation and combined with the existing query.

This already indicates the next issue, namely the realisation of a multi search back-end approach as implemented in PatViz. Here the definition of selection semantics is obviously not sufficient since the information via which back-end service



**Fig. 11.6** Iterative query refinement through analysis of patent result sets

the data can be retrieved is missing. A view such as the tag cloud showing the patent set's most important terms as well as the patent text fields represented in a detail view are related to key word retrieval. Similarly the details dragged from the world map view or the applicant bar chart can only be attached to a new or existing query part addressing the relational back-end where bibliographic information is available from. Interesting semantic relations or images identified in the corresponding detail views can be attached to query parts, addressing semantic search or image similarity search accordingly. Every view must therefore also provide information about which visual aspect can be retrieved from which back-end. In most cases this is a one-to-one relationship where each view is associated with one back-end service. The information on whether the query should be widened or restricted by insight integration can, however, only be supplied by the users themselves. While it is possible to indicate during interaction—by dragging a term or other constraint to a specific location in the query to be modified, or by choosing from context menu whether it should be attached via an AND or an OR—it is very difficult to realise this for all Boolean operators and search back-end specific modifiers. In PatViz, for instance, an operator such as Boolean NOT has therefore to be added explicitly by the user in a second step. While the selection semantics determine the initial constraint of the visual insight being moved to the query, these can, of course, still be changed afterwards according to a user's intention by manipulating the query visualisation directly.

The described interactive transport mechanism also works for multi-selections. The provided selection management tool enables users to transport insights for query refinement found through complex and combinatorial analysis (see Fig. 11.6). An important requirement that must be met by systems allowing for iterative insight integration is that filtering complexity does not exceed the expressiveness of query back-ends, regardless of whether multiple back-ends are employed or not.

## 11.2 Exploiting Interactive Visualisation for Collaboration

Collaborative visual analysis, including related topics such as result presentation, teaching analytic tasks, analytic accountability and many more, is also a distinguished research topic in visual analytics. Interesting works such as [30] and [31] have been proposed in recent years. Since collaborative approaches are useful within a variety of patent analysis tasks, this section presents some of these collaboration aspects.

The described approach taken in PatViz also addresses some aspects of collaborative patent analysis. Every analysis cycle is recorded in order to enable analysts to access previous findings and to perform undo steps if a hypothesis turns out to be not successful regarding the aims of an analytic task. This feature can be exploited in collaborative scenarios, since single improvements of the query are easily traceable for other analysts. Moreover, the described selection management tool offers implicit provenance information for each cycle, because the graph is stored together with all corresponding visual perspectives. If the selection management tool is used for the analysis, single steps, such as the selection of certain information from the views as well as their combination leading to potential reintegration into subsequent queries, can be easily followed.

The queries stored in each iteration encode the (intermediate) results of a whole analytic step in a formal manner. Furthermore, the visual query builder enables users to create, save and reuse parameterisable queries. A parameterised query is set up as a template query. Variables can be applied in those parts that should be configurable and have fixed positions once the query is saved. Upon loading the query, users are prompted to provide a valid input value for each parameter. This mechanism allows for the creation of forms that represent a good starting point for an initial query targeting a special patent search task or that at least support users with templates and basic strategies for common search objectives.

Depending on the scenario in which collaboration tasks should be accomplished, a variety of different collaboration strategies, software and hardware set ups are interesting. Typically, dimensions such as synchronous/asynchronous collaboration and distribution/collocation working set ups are differentiated here. PatViz supports asynchronous and distributed cooperation tasks, since it was developed using standardised communication means, such as web services and platform-independent visualisation techniques. The usage of high-resolution displays and multi monitor set ups [32] is of advantage for single users, because it allows for the efficient exploitation of many different visual perspectives in parallel. Similar set ups are common in stock trading applications, business intelligence, etc. PatViz, however, can be adapted to support collaboration among different persons working together in the same location. Figure 11.7 shows the PatViz user interface running on a high-resolution screen where several persons can inspect, analyse, discuss and present patent information collaboratively. This approach is quite common in strategic command centres, emergency response centrals and other highly collaborative scenarios. In collaborative visual analysis, high-resolution displays provide the further advantage of showing a high number of visual primitives in parallel. This diminishes the



**Fig. 11.7** The PatViz desktop on a large back projection display during a collaborative analysis session

need for extensive information aggregation and offers the possibility to show large amounts of information at once. Such approaches have been researched for fields such as telecommunication and computer network analysis [33]. However, interaction with large displays, especially when working collaboratively, poses a variety of challenges regarding interaction [34].

### 11.3 Discussion and Outlook

In this chapter several aspects of interaction and visualisation in the context of patent analysis have been discussed and possible solutions have been suggested using PatViz as an example. It should be mentioned that integrating interaction in visual perspectives for analysing patent information introduces a number of issues: from the selection of interaction gestures and suitable metaphors, layouts, and presentation forms of visualisation to the choice of suitable selection semantics, to foreseeing the integration of a variety of different views and their meaningful linking to the integration of retrieval services. Clearly, the complexity increases from a technical as well as from a logical point of view in comparison to classic, mainly textual or table-based user interfaces. Another important aspect to be emphasised are the users and their ability to understand graphically presented information and to learn how to interact with it in order to benefit from the available analytic paths. The interaction methods suggested in this section are often implicit, since no obvious interaction elements such as buttons or menus are provided, and not all of the visual means are self-explanatory without previous learning. Evaluation of the PatViz system showed that analysts are not used to interacting with graphical perspectives in order to fulfil their tasks and training is needed before they can use the system efficiently. However, all of them acknowledged the benefit of the analytical power and flexibility these techniques provide. Considering the time and effort it takes to become a specialist in patent retrieval and taking into account the community of experts, who built their patent analysis competence using available tools, existing systems should

not be replaced by interactive visual methods; however, they should be evolved by integrating interactive visualisation to support analytic loops on different levels.

Visual analytics methods are a very powerful means for patent analysis, since visual feedback can help to perceive and assess search results more quickly, thereby increasing users' trust in the validity of their searches and findings during analysis. This is also of special importance for tuning queries to provide higher recall faster, since classic approaches require browsing the information sequentially, which is costly in terms of time. In this case, comparing subsequent results is a promising possibility to observe the effects of modified queries in less time. What has been presented in this chapter is only a first step in this direction. Many other improvements can be realised employing interactive visualisation. For example, the usage of non-Boolean retrieval can be leveraged, thus making it easier to compare and judge the results of several search approaches in parallel. Visual analytics can also enable non-specialist users to exploit clustering methods, machine learning techniques and natural language processing tasks more successfully. Here, visual perspectives can support users by displaying the outcome of such (semi)automated approaches, while interaction methods can be supplied to give feedback in order to recluster, retrain or parameterise them. Applying such strategies can relieve the users of the burden to understand the complexities of these automatic methods by letting them manipulate the underlying mechanisms simply through interacting with visualisations. Promising works in this direction have been published in [35] and [36].

In the development of systems and tools for interactive patent analysis, the main goal is not the invention of new sophisticated visualisations; rather the well-known and established methods have to be combined in an intelligent and seamless manner. The research field of visual analytics develops methods, which achieve the transition from static textual displays to graphical interactions for patent analysis. From the authors' point of view, it can therefore be beneficial to trace the developments in visual analytics closely in order to integrate them in future analysis tools tailored to the patent domain.

## References

1. Tufte ER (1986) *The visual display of quantitative information*. Graphics Press, Cheshire
2. Ware C (2000) *Information visualization: perception for design*. Morgan Kaufmann, New York
3. Card SK, Mackinlay JD, Shneiderman B (1999) *Readings in information visualization: using vision to think*. Morgan Kaufmann, New York
4. Thomas JJ, Cook KA (2005) *Illuminating the path: the research and development agenda for visual analytics*. National Visualization and Analytics Center
5. Keim D, Kohlhammer J, Ellis G, Mansmann F (eds) (2010) *Mastering the information age: solving problems with visual analytics*. Eurographic Association, Goslar
6. Chang R, Lee A, Ghoniem M, Kosara R, Ribarsky W, Yang J, Suma E, Ziemkiewicz C, Kern D, Sudjianto A (2008) Scalable and interactive visual analysis of financial wire transactions for fraud detection. In: *Information visualization, 2008* (7)
7. Stasko J, Görg C, Liu Z (2008) Jigsaw: supporting investigative analysis through interactive visualization. In: *Information visualization, 2008* (7)



8. Ware C (2004) *Information visualization: perception for design*, 2nd edn. Morgan Kaufman, San Mateo
9. Börner K, Chen C (2002) *Visual interfaces to digital libraries*. Lecture notes in computer science, vol. 2539. Springer, Berlin
10. Börner K, Chen C, Boyack K (2003) *Visualizing knowledge domains*. *Annu Rev Inf Sci Technol* 37
11. Shneiderman B (1996) The eyes have it: a task by data type taxonomy for information visualizations. In: *Proc IEEE symp visual languages*
12. Sears A, Jacko JA (2007) *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, 2nd edn. CRC Press, New York
13. Trippe AJ (2002) *Patinformatics: identifying haystacks from space*. *Searcher* 10(9)
14. Trippe AJ (2003) *Patinformatics: tasks to tools*. *World Pat Inf* 25(3):211–221. doi:[10.1016/S0172-2190\(03\)00079-6](https://doi.org/10.1016/S0172-2190(03)00079-6)
15. Yang YY, Akers L, Klose T, Barcelon Yang C (2008) *Text mining and visualization tools—impressions of emerging capabilities*. *World Pat Inf* 30
16. Moehrl MG, Walter L, Bergmann I, Bobe S, Skrzypale S (2010) *Patinformatics as a business process: a guideline through patent research tasks and tools*. *World Pat Inf* 32(4):291–299. doi:[10.1016/j.wpi.2009.11.003](https://doi.org/10.1016/j.wpi.2009.11.003)
17. Card S, Pirolli P (2005) *Sensemaking processes of intelligence analysts and possible leverage points as identified through cognitive task analysis*. In: *International conference on intelligence analysis*
18. Chen Y, Yang J, Ribarsky W (2009) *Toward effective insight management in visual analytics systems*. *Pac Vis*. doi:[10.1109/PACIFICVIS.2009.4906837](https://doi.org/10.1109/PACIFICVIS.2009.4906837)
19. Koch S, Bosch H, Giereth M, Ertl T (2009) *Iterative integration of visual insights during scalable patent search and analysis*. *IEEE Trans Vis Comput Graph*. doi:[10.1109/TVCG.2010.85](https://doi.org/10.1109/TVCG.2010.85)
20. Wanner L, Baeza-Yates R, Brüggmann S et al. (2008) *Towards content-oriented patent document processing*. *World Pat Inf* 30(1)
21. Inselberg A, Dimsdale B (1990) *Parallel coordinates: a tool for visualizing multi-dimensional geometry*. In: *IEEE visualization*
22. Roberts J (2007) *State of the art: coordinated & multiple views in exploratory visualization*. In: *Proc 5th int'l conf coordinated and multiple views in exploratory visualization*
23. Elmqvist N, Stasko J, Tsigas P (2008) *DataMeadow: a visual canvas for analysis of large-scale multivariate data*. In: *Information visualization*
24. Spoerri A (1994) *InfoCrystal*. In: *Proc SIGCHI conf human factors in computing systems*. ACM, New York
25. Fishkin K, Stone MC (1995) *Enhanced dynamic queries via movable filters*. In: *Proc SIGCHI conf human factors in computing systems*. ACM Press, New York
26. Stolte C, Tang D, Hanrahan P (2002) *Polaris: a system for query, analysis, and visualization of multidimensional relational databases*. *IEEE Trans Vis Comput Graph*
27. Hearst MA (2009) *Search user interfaces*. Cambridge University Press, New York
28. Wirth N (1973) *Systematic programming: an introduction*. Prentice Hall PTR, Englewood Cliffs
29. Ahlberg C, Williamson C, Shneiderman B (1992) *Dynamic queries for information exploration: an implementation and evaluation*. In: *Proc ACM CHI'92 conf human factors in computing systems*
30. Richter Lipford H, Stukes F, Dou W, Hawkins ME, Chang R (2010) *Helping users recall their reasoning process*. In: *Proc IEEE symp visual analytics science and technology*, pp 187–194. doi:[10.1109/VAST.2010.5653598](https://doi.org/10.1109/VAST.2010.5653598)
31. Callahan SP, Freire J, Santos E, Scheidegger CE, Silva CT, Vo HT (2006) *VisTrails: visualization meets data management*. In: *Proc ACM SIGMOD int'l conf management of data*
32. Czerwinski MP, Smith G, Regan T, Meyers B, Robertson GG, Starkweather G (2003) *Toward characterizing the productivity benefits of very large displays*. In: *Interact*
33. Wei B, Silva CT, Koutsofios E, Krishnan S, North S (2000) *Visualization research with large displays*. *IEEE Comput Graph Appl*

34. Czerwinski M, Baudisch P, Meyers B, Robbins D, Smith G, Tan D (2005) The large-display user experience. *IEEE Comput Graph Appl*
35. Crossno PJ, Dunlavy DM, Shead TM (2009) LSAView: a tool for visual exploration of latent semantic modeling. In: *IEEE symposium on visual analytics science and technology*
36. Paulovich FV, Minghim R (2008) HiPP: a novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Trans Vis Comput Graph*

## Part IV

# Classification

As noted in the introductory chapter by Alberts and colleagues, almost all patent documents are classified according to the International Patent Classification (IPC) and possibly also one or two other classification schemes, like the United States Patent Classification (USPC) or the European Patent Classification (ECLA). These classification systems are hierarchical, detailed and extensive. For example, the IPC has over 60,000 subgroups (the finest level of classification) arranged in eight sections, corresponding to major areas of technology.

This wide availability of classification has many implications for the operation of the patent system. Many professional patent searchers make extensive use of classification codes to restrict and narrow their searches. The patent offices put very significant effort into ensuring that patents are classified in a comprehensive but accurate way: they are very interested in means of automating the task or assisting examiners with the classification process. Technical opportunities arise for IR and Machine Translation researchers and technologists, for example to utilise the classification of a patent when attempting to automatically acquire new vocabulary.

The challenge here is two fold: First, search technologists must find ways to assist the patent applicants and the patent offices to classify their documents. Second, they must find ways to leverage the classification of patent documents to improve the automated parts of search systems, going beyond merely restricting result sets to particular patent classes. For example this might include recognising in the automatic indexing cycle that the same term means different things in different areas of technology.

This part is comprised of three chapters about patent classification. Benzineb and Guyot start the part with a chapter on the issues, utility and state-of-the-art of Automated Patent Classification tools. Then, Koster and colleagues go into uncharted territory and introduce a new way to look at documents, beyond the usual bag-of-words model, and apply it to identify the ‘aboutness’ of a document, and relate that to the classification of a patent. The final chapter, by Harris and colleagues, focuses on the use of patent classification data in a highly automated patent prior art search system, and formally show what was perhaps known, but never proven: that using classification in the search process helps, and that different classification systems have different utilities in the search process.

# Chapter 12

## Automated Patent Classification

Karim Benzineb and Jacques Guyot

**Abstract** Patent classifications are built to set up some order in the growing number and diversity of inventions, and to facilitate patent information searches. The need to automate classification tasks appeared when the growth in the number of patent applications and of classification categories accelerated in a dramatic way. Automated patent classification systems use various elements of patents' content, which they sort out to find the information most typical of each category. Several algorithms from the field of Artificial Intelligence may be used to perform this task, each of them having its own strengths and weaknesses. Their accuracy is generally evaluated by statistical means. Automated patent classification systems may be used for various purposes, from keeping a classification well organized and consistent, to facilitating some specialized tasks such as prior art search. However, many challenges remain in the years to come to build systems which are more accurate and allow classifying documents in more languages.

### Abbreviations

AI	Artificial Intelligence
APC	Automated Patent Classification
ECLA	European Patent Classification
EPO	European Patent Office
IPC	International Patent Classification
kNN	$k$ Nearest Neighbors
MCD	Master Classification Database
NN	Neural Network
PCT	Patent Cooperation Treaty
SVM	Support Vector Machine
WIPO	World Intellectual Property Organization

---

K. Benzineb (✉) · J. Guyot  
SIMPLE SHIFT, Ruelle du P'tit-Gris 1, 1228 Plan-les-Ouates, Switzerland  
e-mail: [karim@simple-shift.com](mailto:karim@simple-shift.com)

## 12.1 Introduction

An efficient way to facilitate the retrieval of objects is to arrange them beforehand according to an order, which makes sense for most of the people who will do the searching. A good illustration is the way books are organized on the shelves of a library: they are generally grouped by subject, so that the searcher is easily oriented toward the relevant area. A side advantage of this method is that reaching to any specific book automatically provides you with more information on the same topic.

Such is the purpose of classification. The more information you have to manage, the more structured and detailed your classification should be to allow for easy navigation and precise search.

This chapter is meant to explain how classification supports patent management and retrieval, why it is useful to automate it, and how this may be done. Automated patent classification (hereafter APC) has several objectives, which are defined in Sect. 12.2 below. Two of the most widely used patent classifications, namely IPC and ECLA, are reviewed in Sect. 12.3. The structure and content of patent collections used to build automated classification systems are described in Sect. 12.4. Selected algorithms and tools on which classification software is often developed are explained in Sect. 12.5. Various evaluation approaches of APC are suggested in Sect. 12.6, Use Cases are presented in Sect. 12.7, and the main challenges of APC in the close future are discussed in Sect. 12.8.

## 12.2 Definition and Objectives of Automated Patent Classification

### 12.2.1 Definition

Automated patent classification may be defined as the process by which a computer suggests or assigns one or several classification codes to a patent on the basis of the patent's content. This definition implies that several conditions must be satisfied:

- a taxonomy, i.e. a patent classification in which each category is clearly defined and has a unique code, must previously exist;
- a full collection of patents previously classified by humans under that classification must be available to train and test the system;
- the content of the patent to be classified (text and possibly pictures, graphs, etc.) must be in electronic format so as to allow for computer processing.

Although these consequences appear to be trivial, they place a heavy constraint on the very possibility to build an automated classifier because one of the most difficult parts is generally to find or build a training set of patents, which is large and well distributed enough.

## 12.2.2 Objectives

The overall objective of patent classification is to assign one or several category codes to a patent, which is not categorized yet in a given patent classification system. The objective of automating the process is to make it much faster and more systematic than the human process, thus saving time and costs.

- **Faster:** Human examiners must read the whole patent text, than browse the classification categories which seem most relevant to them, and finally make one or several choices. The whole process can take up to several hours, while a computer performs the same task in a matter of milliseconds.
- **More systematic:** Human classification may be subjective because it is based on an individual examiner's judgment (which in turn depends on his/her education, experience and mindset) and because it is entrusted to a large number of examiners (up to several thousands in some Patent Offices).

Automated classification systems tend to make the same choices under the same conditions; this leads to more harmonized results. Beyond this immediate objective, APC has in fact a deeper purpose, which is twofold: it has an organizational mission and it must facilitate search tasks.

### 12.2.2.1 Organization

The essential process of classification is tagging, i.e. assigning a code to an element, which must be classified. In the case of patent classifications, the number of codes to choose among may be extremely high: The International Patent Classification (IPC) has over 60,000 categories and the European Patent Classification (ECLA) has about 129,000.

Besides, the number of patent applications to classify is also very large and it keeps growing: according to statistics from the World Intellectual Property Organization (WIPO), over 1,850,000 patents applications were filed in 2007, up from about 926,000 in 1985.

APC's organizational mission is to assign to those patent applications a classification code in order to preserve the consistency of an order which was defined by human experts. It puts patents "where they belong". This can be done for new incoming patents, but also backwards on previously categorized patents in order to re-arrange them when the classification was modified (this is called "re-classification").

As a side product of this role, APC also allows building so-called "pre-classification" systems: In a large patent organization, an APC system can read an entering patent application and route it to the relevant team of experts who will be in charge of making a decision about its final categorization.

### 12.2.2.2 Search for Prior Art, Novelty, etc.

A major function of APC is to support patent search. Typical goals of a patent search include prior art (patentability), novelty, validity (legal status), freedom to operate, infringement search, etc.

A major issue in patent search is the size of the search space: There are about 40 million patents in WIPO's Master Classification Database (MCD), and this does not represent all of the world's patents in all languages.

The objective of APC in terms of search support is twofold.

- Reducing the search space: By proposing one or several classification codes for a patent application, APC allows to focus the search on the most relevant patent categories, thus excluding most of the patent search space.
- Allowing for search on the basis of similarity: Since the vast majority of patents were classified manually by human examiners, some patents may not be categorized under the expected codes. In order to extend a search to other categories (e.g. for prior art search), APC allows comparing the content of the patent application with the content of each patent in the training set. It may thus retrieve patents which were not classified in the same category as the patent application, but whose content is very similar to it.

It should be underlined here that APC systems are not only used by patent practitioners such as inventors and patent attorneys. Many organizations use them for other purposes such as technological watch, economic intelligence, etc.

### 12.2.3 Historical Factors

Classifying patents became necessary because of a fast growth in both the number of patents and the number of patent fields.

Logically, the need to *automate* patent classification resulted from the same factors when they reached a higher scale. In particular, the fast growing number of patent applications mentioned above was a driving factor of research on automated tools.

Additional factors also played a role, in particular the fast-increasing number of human examiners, which was (and still is) leading to classification consistency issues. Besides, the hyper-specialization of patent categories made it impossible to entrust the classification job to "universal experts"; it called for a specialization of the examiners themselves, which in turn provoked a diversification of the classifying methods, criteria—and results.

This situation is further complicated by the fact that some patents are of horizontal nature, i.e. they can or should be classified in several categories. For example, a tobacco humidifier can be linked to industrial processes, to storing processes and to agricultural products. Multi-category classification made it even more complex to categorize and to retrieve similar patents, and called for some mechanical help.

## 12.3 A Few Words about Patent Classifications

Patent classifications, or taxonomies, generally come in the form of a hierarchy of categories: the top level includes very broad categories of inventions, so the number of top categories is very small. The second level includes more narrow categories, the third level even more precise categories, and so on. Thus as we go down the taxonomy levels, the number of categories grows dramatically.

Two patent taxonomies are briefly considered below: the International Patent Classification (IPC), which is built and maintained by the World Intellectual Property Organization (WIPO), and the European Patent Classification (ECLA), which is built and maintained by the European Patent Office (EPO). Both of them are available online on the respective organization's website.

### 12.3.1 IPC

The IPC (Edition20090101) is divided into a Core and an Advanced Level; the Core Level goes from Section down to Main Group, with some technical sub-groups. The Advanced Level contains all the sub-groups of the IPC. According to WIPO, “the core level is intended for general information purposes, for example, dissemination of information, and for searching smaller, national patent collections. (...) The advanced level is intended for searching larger, international patent collections.”

At the Advanced Level, the IPC has the following tree structure: eight Sections, 129 Classes, 639 Sub-Classes, 7,352 Main Groups and 61,847 Sub-groups.

The sections (top categories) are the following: Section A—Human Necessities; Section B—Performing Operations; Transporting; Section C—Chemistry; Metallurgy; Section D—Textiles; Paper; Section E—Fixed Constructions; Section F—Mechanical Engineering; Lighting; Heating; Weapons; Blasting; Section G—Physics; Section H—Electricity.

This top level illustrates the essential challenge of any patent taxonomy: it has to describe the world, and it must be able to include objects and ideas, which, by definition, were never thought of before. Thus it has to be as general and open as possible. For that reason it does not make much sense to classify patents at the Section level. Even the Class and Sub-Class levels are often considered too wide to be useful for professionals (examiners, patent attorneys, etc.). Therefore automated classification systems are generally required to categorize patents at least at the Main Group level. This means any such system should at least be able to manage over 7,000 categories; it also means the system must support a large number of patent examples for the training phase (see Sect. 12.5), since each category must have at least a few example documents for the system to correctly identify it.

At first glance, this enormous quantity of data makes the field of patent classification particularly fit for computerized statistical processing. However, history, while bringing about the reasons for automating patent classification, also produced complicating factors, which actually hamper the efficiency of computerized processing.



These factors are essentially linked to exceptions to the general classification rules: many patent categories contain one or several notes, which indicate that specific types of inventions should actually be classified somewhere else. For example, the category A23 in the IPC has the following title: “Foods or foodstuffs; their treatment, not covered by other classes”. This initially requires knowing which treatments are “covered by other classes”. But additionally, category A23 includes the following notes:

“Note(s)

Attention is drawn to the following places:

C08B: Polysaccharides, derivatives thereof

C11: Animal or vegetable oils, fats, fatty substances or waxes

C12: Biochemistry, beer, spirits, wine, vinegar

C13: Sugar industry.

Processes using enzymes or micro-organisms in order to: liberate, separate or purify a pre-existing compound or composition, or to treat textiles or clean solid surfaces of materials are further classified in Sub-Class C12S.”

The very human nature of a patent taxonomy and of its evolution therefore makes it very difficult to define systematic classification rules and build them into a model; categories are best described by the documents they contain. This is why example-based training technologies tend to be favored to build automated patent classifiers.

### ***12.3.2 ECLA***

The ECLA taxonomy is worth mentioning in addition to the IPC because it is a kind of extension of the IPC: It is identical to the IPC down to Main Group level, but it is more detailed at Sub-Group level, where it contains 129,200 categories, thus allowing for a finer-grain classification.

While the IPC seems to be more oriented toward the publication of patents, ECLA is rather more focused on supporting patent information search in the context of a patent application. It is extensively used, for example, by the EPO examiners in their daily work. ECLA is also used to classify the PCT’s minimum documentation and other patent-related documents, including utility models.

According to the EPO, 26.2 million documents had an ECLA class in 2005. Combined with the 129,200 categories, this gives a broad idea of the “classification space” which must be managed by any automated classifier.

## 12.4 Patent Collections

### 12.4.1 Structure of a Patent

The structure of a patent is important because the precision of an APC system directly depends on the quality of the training data, which in turn means that the content to be provided as training material must be carefully chosen. Although there are many ways of representing the structure of a patent (with more or less information details), the content of most patents is organized in the following way.

- The bibliographic data: the patent ID number, the names of the inventor and the applicant, the title of the patent, and the abstract.
- The claims, in which the applicant explains what the invention is made of and which application fields the patent is sought for.
- The full text, which contains the complete description of the patent.

Other fields may be found, such as the agent's name, priority data, publication and filing languages, etc. It is also frequent, for example in the fields of chemistry or mechanics, to find graphics or other types of illustrations. The fields are generally represented in an XML structure, which may look like this:

```
<record cy="WO" an="SE0001823" pn="WO012189020010329"
dnum="0121890" kind="A1"> [Unique patent number, which can include
the date]

<ipcs ed="7" mc="D21H01120"> [This is the IPC classification number]
<ipc ic="D21H01725"></ipc>
</ipcs>

<ins> [Inventors]
<in>LINDSTRÖM, Tom</in>
<in>GLAD-NORDMARK, Gunborg</in>
<in>RISINGER, Gunnel</in>
<in>LAINE, Janne</in>
</ins>

<pas> [Patent Applicant]
<pa>STFI</pa>
</pas>

<tis> [Title]
<ti xml:lang="EN">METHOD FOR MODIFYING CELLULOSE-BASED
FIBER MATERIAL
</ti>
</tis>
```

```

<abs> [Abstract]
<ab xml:lang="EN">A method for modifying cellulose fibers, which are
treated for at least 5 minutes with an aqueous solution of CMC or CMC-
derivative (...)
</ab>
</abs>

<cls> [Claims]
<cl xml:lang="EN"> Claims
1. Method for modifying cellulose fibers, characterized in that the cellulose
fibers
are treated for at least 5 minutes with an aqueous electrolyte-containing solu-
tion (...)
</cl>
</cls>

<txts> [Full Text Description]
<txt xml:lang="EN">
Method for modifying cellulose-based fiber material
This invention concerns the technical field of paper manufacture, in particular
chemical (...)
</txt>
</txts>
</record>

```

Our experience showed that most of the time, only a part of this content should be used to feed an automated classifier. First, some data have a higher classifying power: it may be the case, for example, of the inventor's name, because inventors tend to invent in a specific field. The applicant's name is also important because it is often a company with a specific area of expertise (although large companies may apply for inventions in various fields). Second, a field such as the Claims may be of little interest for training purposes because it was deliberately written in a vague style so as to cover the widest possible application area. Words in the Claims section tend to be ambiguous and do not help the classifier to make a decision (our experience showed for example that a classifier often has a higher accuracy when trained on the Abstract than on the Claims section). Third, most automated classifiers (and in any case the classifiers based on the algorithms described in Sect. 12.5 below) are exclusively based on text and cannot make use of any graphic information. This means for example that some categories such as Chemistry or Mechanics, whose descriptions heavily rely on diagrams, are not so well classified by text-only tools. Finally, text-based learning machines can get saturated beyond a given number of words: adding more and more words in each example actually ends up creating noise and confusing the machine, which drives the classification precision down. In fact, we found that it is generally more efficient to train an APC system on a large number of small examples for each category than on a small number of large documents.

Therefore the fields which tend to be preferred (through empirical findings) as training material for APC are essentially the bibliography fields, namely the inventor, applicant, title and abstract. However, it was observed that adding some information from the full text description does improve the classification precision, provided that the full text is truncated (our experience suggested the limit of 350 or 400 different indexed words) so as to avoid the saturation issue.

### ***12.4.2 The Distribution Issues***

Creating a classification inherently creates distribution imbalances. In the case of patent classifications, those imbalances are essentially found in the distribution of example documents (patents) across the categories, and in the distribution of words within a patent.

#### **12.4.2.1 Distribution of Example Documents: The Pareto Principle**

Building a patent collection with regards to a classification amounts to separating patents according to *external* criteria: patents are not grouped because of intrinsic properties (such as the number of words they would share, for example) but because they address a topic which was defined externally by human experts, with regards to their own “knowledge of the world”.

When groups of objects are separated according to external criteria, they tend to show a Pareto-like distribution across the categories, i.e. over a large volume of categories and documents, about 80% of all the documents are classified in about 20% of the categories. This creates a structural issue for any artificial intelligence system, which is based on example-based learning. Most automated patent classifiers belong to this family of tools: they need to be trained on typical examples of each category to be able to correctly identify those categories later on.

If some categories are poorly documented, i.e. they have little or no typical patents to feed the computer with, they are very unlikely to be ever predicted by the system because it will never be able to identify a patent typical of such a category. A distribution reflecting the Pareto Principle means that although 20% of the categories will be well documented, the remaining 80% will share only 20% of the total training set. Inevitably, many of those categories will “disappear” in the training process. Solutions to this issue are considered in Sect. 12.5 below, but this remains one of the core problems of any automated classification system.

#### **12.4.2.2 Distribution of Words in a Patent: Zipf’s Law**

APC systems depend heavily on the words, which are contained in a patent. Neural network applications, for example, give a weight to each word with regards to each

category, depending on whether it appears more frequently in one category than in the other ones. The presence, in the text of a patent, of a large number of words which are heavily weighted in favor of a given category drives the application to assign that category code to the patent.

Since the weighting is roughly performed according to the number of times a given word appears in the examples of a category, the system would work better if most of the words appeared frequently in the training documents: it would be easier for the computer to compare their respective occurrences in each category. The problem is that the number of very frequent words is very small, and most words occur in fact rather rarely. This situation is described by Zipf's law, which is well explained on Wikipedia: "The frequency of any word is inversely proportional to its rank in the frequency table". In other words, "the most frequent word occurs approximately twice as often as the second most frequent word, which occurs twice as often as the fourth most frequent word, etc."<sup>1</sup>

Zipf's law tells us that most of the words encountered by the APC system will actually not occur very frequently, so the system will have to work on rather rare words. Moreover, the combination of Pareto's Law and Zipf's Law suggests keeping in the index words, which only occur a small number of times (e.g. four or five times) because they could be representative of a class, which is poorly documented.

### ***12.4.3 The Language Issues***

Automated classification systems are met with two major issues when trained on patents: one is linked to natural language in general, and the other one to the particular tongue used in the patent.

#### **12.4.3.1 Natural Language Issues**

The ambiguity of natural languages is a well-documented problem, which has a major impact on information retrieval in general, and on automated classification in particular. We will focus here on the issues specifically linked to classification.

Most APC systems use example-based training, i.e. they "read" the content of the various texts provided as typical examples of a given category in order to correctly identify that category. The system is not able to distinguish the various meanings of ambiguous words (polysemy). As a result, it will consider that it is always the same word, which reduces the quantity of information available to identify specific categories.

Another issue with natural languages is linked to semantics: the fact that a word like "not" or "without", which may reverse the meaning of a sentence, is probably

---

<sup>1</sup>[http://en.wikipedia.org/wiki/Zipf's\\_law](http://en.wikipedia.org/wiki/Zipf's_law). Accessed 23 Dec 2010.

going to be ignored because it is frequent and thus the invention may be classified in a category of things it does *not* do.

A different kind of problems is also met with the various peculiarities of human languages, such as the so-called “collocations” or “compound words”, i.e. expressions which are composed of more than one word and whose collective meaning is different from the added meaning of each individual word. For example, an “electric plug” is very different from a “spark plug”. Automated classifiers, when used without any linguistic processing, index each word separately and thus lose this collective meaning.

There are many other difficulties linked to linguistics, such as inflections, agglutination (some languages like German stick together various words to build a new entity), or segmentation (choosing the correct number of ideograms which constitute a word in Asian languages), etc.

The issues described above call for linguistic processing, but this may be a costly improvement because it is different for each language (so a linguistic system must be built for each working language) and it may slow down the program’s execution.

There are other language issues, however, which may not be solved by linguistic tools, but more probably by statistic processing. The most important one is probably the growing vocabulary extent in each category: new applicants file new patent applications over time, and each of them uses his/her own vocabulary to describe the invention. The underlying issue here is linked to the compositional nature of human language: by combining different words (synonyms) it is possible to say the same thing in many different ways. Thus the number of words typical to a given category grows over time, and a growing number of those words tend to be found in a growing number of categories.

Overall, it should be stressed that technologies such as neural networks (see Sect. 12.5) and others allow one to represent the global context of a patent, which is an efficient solution to get rid of the ambiguity issue in natural languages. An isolated word may have several meanings, but its frequent association with other non-ambiguous words helps a computer to differentiate the various uses of that word. Additionally, in the specific case of neural networks, if a word is so ambiguous that it may be found often and in very diverse situations, the weight of this word will become so low that the word will eventually be discarded for classification purposes.

#### 12.4.3.2 The Corpus Language Issue

APC systems are trained on previously classified examples to recognize patterns of words, which are typical of a given category. However, a system which is trained on English may only classify new patents in English.

Classification (as well as information retrieval) in foreign languages, and more particularly in Asian languages (above all Chinese, Japanese and Korean), are services which tend to be increasingly required by patent applicants, patent attorneys and many other patent professionals and organizations. However, it is still difficult to find large training sets in these languages, mostly for one or several of the following reasons:

- only a small number of patents were filed in the language considered, both at the national and international levels (some countries have a small number of inventors and they are specialized in a small number of fields);
- a number of patents are available but they did not originally exist in electronic form, so only an image scan is available. In this situation the patents may only be used if a good Optical Character Recognition (OCR) software exists for the language considered;
- large training sets were compiled by a private or public organization, but they are kept private or they are sold at a cost which is prohibitive;
- patents are available in a reasonable quantity and in electronic form but they are not classified under international classifications such as the IPC or ECLA so there is a problem to build the initial training set (this is known as the “bootstrap” problem).

For those countries whose number of filed patent applications is quickly growing, training corpuses will soon be available. For the other ones, should an automated patent classifier be necessary, various solutions may be considered, in particular machine translation.

### 12.4.3.3 The Time Issue

Patent classifications obviously evolve over time: new categories are added while old ones may become deprecated, and some categories may be merged together or broken down in finer ones. However, time also has a direct effect on the language used in the patents. First, the vocabulary of any given category may change over time; this is in particular the case in the field of computer science, where new technologies and standards frequently drive a terminological evolution. Second, the creation of new categories may increase polysemy, as some existing words (like “cookie”) are being re-used in new contexts (“Internet cookie”). Finally, the very definition of a category may evolve over time; this issue is known as the “topic drift”. This may happen for example when a traditional field (such as printing) is slowly being changed by the introduction of new technologies (in this case, IT): the application or result of the new patents is still directly linked to printing, but the domain itself becomes much wider.

## 12.5 State-of-the-Art Technologies

Many algorithms may be used for the purpose of automated classification. Most of them come from the world of artificial intelligence (AI) and have been known for several decades (sometimes more); they were revived over the past decade because the spectacular progression in CPU and RAM capacities allowed one to perform more and more calculations in a decreasing time and at a decreasing cost. A general review of most technologies used in the field of APC can be found in [1].

We will focus here on three algorithms, which are among the most frequently used in the field of automated classification, namely Neural Networks (NN), Support Vector Machine (SVM) and the  $k$  Nearest Neighbors ( $k$ NN). Other technologies such as Bayesian algorithms, the Rocchio method or Decision rules are also interesting in specific cases, but a complete review of existing technologies and their merits is out of the scope of this chapter.

### ***12.5.1 Neural Networks***

A neural network is a network of individual values, each value representing the weight of a given word with regards to a given category, i.e. it tells how well the word (called a “feature”) represents the category. Initially the system reads all the documents provided as “good examples” of each category. It compares all the words of all the training documents for all the categories: words which are found too often are given a low weight, because they have a low “classifying power”. Conversely, words which seem to be very typical of a given category are given a higher weight because they are strong discriminators. After the training phase, when the classifier is required to classify a new patent, the patent’s content is turned into a set of words and weights, which are then compared to those in the neural network; the system chooses the category for which the weights are maximized.

Neural networks are currently among the best-performing patent classifiers for several reasons:

- they scale up extremely well, i.e. they support a large classification space (defined as the number of features time the number of categories);
- they can be combined, so in a tree hierarchy such as the IPC or ECLA, a large number (over a thousand) of neural networks may be built and connected at each level, thus allowing users to ask for a direct classification at any level of the tree;
- after the training phase, the resulting neural networks can be saved for later querying, so the system can reply extremely quickly to users;
- neural networks are trained on strings of characters, so they can be used to classify any type of symbols (e.g. any language, but they can also be used to classify DNA strings, etc.).

### ***12.5.2 SVM***

The Support Vector Machine, like the Neural Networks, is a system which has to be trained beforehand, but unlike the NN, it does not assign weights to words; the words are considered as dimensions of a space, and each example of a category is considered as a point in this space. SVM tries to find the plane surface, which separates two categories with a gap as wide as possible.



SVM is interesting to use because it is very accurate: in fact its capacity to build separations between categories is higher than that of the NN. Besides, and unlike the NN, it has no internal configuration parameters, so it is easier to use. However, it does not seem to be widely used to classify patents at the lowest levels of large classifications such as the IPC or ECLA, probably because it only supports a small combination of words and categories, and it is very slow to train.

### ***12.5.3 kNN***

The  $k$  Nearest Neighbor algorithm compares a document to be classified to a number of other, previously classified documents ( $k$  stands for the number of documents to be compared). Similarities between documents are computed by comparing word distributions. The category of the new document is calculated with regards to the categories of the neighboring documents by weighting their contributions according to their distance; thus it is also a geometric measure.

Unlike the NN and SVM algorithms, the  $k$ NN does not have to be trained before being used: all the calculations are performed when a document is submitted to the system. For this reason, it is generally not used to predict categories within large classifications such as the IPC or ECLA, because it is considered too slow to reply.

On the other hand, it is very useful for prior art search because it can systematically compare the patent application with the existing patents, retrieve the closest ones and show them to the user.

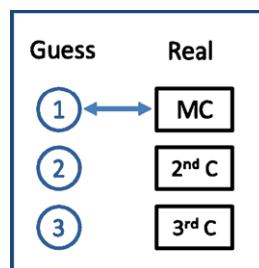
## **12.6 Evaluating Automated Patent Classification**

The accuracy of an APC system is calculated over a test set which was held out from the training set, so the classifier has to categorize patents it has never seen before.

In a neural network system, for example, the classifier is initially trained over patents which were previously classified by human experts. Then a test set is submitted to the classifier, which predicts one or several categories for each test patent. Finally those predicted categories are compared to the correct classes (which are also known since the test set was also previously classified by humans) and an accuracy score is automatically calculated.

### ***12.6.1 Standard Evaluation Methods: Precision, Recall, F1***

In the general field of information retrieval, accuracy scores are often calculated according to one or several of the following three standard methods: a precision score, a recall score and a so-called “F1” score, which is an average of precision

**Fig. 12.1** Top prediction

and recall. Those methods are also valid to assess the accuracy of an automated patent classifier.

According to Wikipedia, “Precision can be seen as a measure of exactness or fidelity, whereas Recall is a measure of completeness.”<sup>2</sup> More specifically, in the field of patent classification, for a given patent submitted to the automated classifier:

- Precision is the number of categories correctly predicted by the classifier divided by the total number of predicted categories;
- Recall is the number of categories correctly predicted by the classifier divided by the total number of existing correct categories (i.e. the number of categories which should have been retrieved).

As for the F1 score (also called F-score or F-measure), it is a weighted average (more precisely the harmonic mean) of precision and recall:

$$F1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

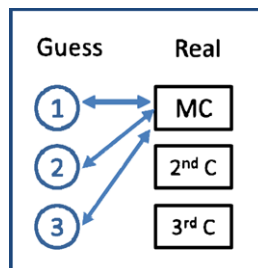
Several variations of the F1 score can be used to place more emphasis on precision or on recall. Choosing the most relevant measure directly depends on the intended use of the search engine or the classifier: sometimes only precision or recall may be looked at. For example, a patent classifier which assigns one or several correct category codes to an incoming patent application may be considered good enough, although it may not have assigned *all* the correct categories. In such a case, only the precision score may be taken into account.

### 12.6.2 Customized Use of Accuracy Measures

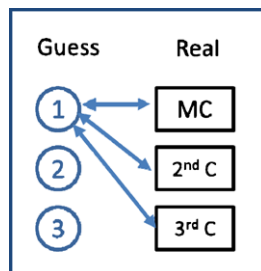
Whether the type of measure is precision, recall or F1, there are still various ways of calculating it. The most common way is to determine whether the top category predicted by the classifier corresponds to the first real category of the patent. This is all the more useful when the classification includes a main category (called “MC” in Fig. 12.1) and several secondary categories:

<sup>2</sup>[http://en.wikipedia.org/wiki/Precision\\_and\\_recall](http://en.wikipedia.org/wiki/Precision_and_recall). Accessed 23 Dec 2010.

**Fig. 12.2** Three Guesses measure



**Fig. 12.3** All Categories method



This accuracy assessment makes sense for fully automated solutions, for example when an APC system receives electronic patent application files and routes them to the most relevant team of human examiners (this application is called “pre-classification”). In this case only the best choice is kept because the system must only route the file to a single team, so the accuracy score should reflect this constraint.

However, other methods of calculating the accuracy may be chosen, for example when the APC system is used by a human expert as a classification assistant. In this case, more than one guess may be accepted by the user, since he/she will make the final decision. On the other hand, in the example of the pre-classification task described above, only one good answer is possible, because only one team is the most relevant to classify the patent. In this situation we can use for instance a “Three Guesses” measure as shown in Fig. 12.2.

In this measure, if the most relevant category appears in one of the top three predictions, the patent is considered to be correctly classified.

Other types of applications can be considered (see Sect. 12.7 below); for example, patents can generally be attributed to multiple categories, so a human examiner who would use the APC as a classification assistant may be willing to accept any good suggestion, even if it is not the main category or if there is no main category (which is the case under ECLA). In this situation the accuracy measure would be calculated according to an “All Categories” method as illustrated in Fig. 12.3.

This last method of calculating the accuracy produces of course the highest scores, since it is more flexible as to what constitutes a correct prediction. It is for example the method used by WIPO to assess IPCCAT’s precision, because IPCCAT is a classification assistant. Its typical precision scores for English patents are about 90% at Class level, 85% at Sub-Class level and 75% at Main Group level.

It should be mentioned that the methods proposed above address the situation where the APC system provides so-called “unsupervised” predictions, i.e. it has no human help such as limiting the classification space to a given section, or refining to a finer category after validating a coarser one. More interactive systems, where the APC system can be guided by a human user, may provide for more accurate predictions, especially at the more detailed levels.

## 12.7 Use Cases

There is a wide range of possible applications for APC systems. The ones that are suggested below have been actually implemented (interactive classification and prior art search) or are being considered (pre-classification and re-classification) in particular by international organizations.

### 12.7.1 *Pre-classification*

Pre-classification, as mentioned earlier, is the task of automating the distribution of incoming patent applications among the various possible groups of examiners. Large patent offices have organized their teams of examiners according to fields of expertise. When a new patent application reaches the patent office, it must be routed to the most relevant group of experts. This can be automated with an APC system.

Such an application has generally an excellent accuracy performance because the number of groups of experts is generally less than 100 (i.e. much less than the number of patent classification categories, for example). However, since the system operates without any human supervision, it is generally required to compute a confidence score for each decision, and the automated routing is only allowed when the confidence score is above a pre-defined threshold.

### 12.7.2 *Interactive Classification*

APC systems can be used as classification assistants for human examiners in an interactive manner: the examiner submits a patent application, the APC system makes one or several predictions at a given level of the classification, and then the examiner can decide to:

- ask for a refined prediction down to a finer-grain level of the classification (for instance at Main Group level after an initial prediction at Sub-Class level);
- ask for another prediction directly at the finer-grain level (e.g. a prediction directly at Main Group level, instead of going first to Sub-Class level—it is important

to understand that in the case of neural network systems, the APC's prediction will not be the same when predicting directly at the lower level and when going through an intermediate level because the APC does not use the same neural networks);

- force a prediction under a given category, for instance by defining that he/she wants only predictions under the A01B Sub-Class.

The World Intellectual Property Organization (WIPO) proposes such an interactive APC tool on its website; the tool is called IPCCAT and it is freely available to the public.

### ***12.7.3 Re-classification***

In large patent classifications such as the IPC or ECLA, some categories grow over time up to the point where they contain too many patents of various content. In that case they must be broken down into several more detailed categories (at the same level). Conversely, some categories may end up with very few or no patents, either because they are too narrow or specialized, or because they were defined through a rather theoretical process.

Thus patent classifications must be re-organized from time to time; for instance, the current version of the IPC is the ninth one, and new versions might be published from now on at least on an annual basis.

Re-classification is the process by which patent categories are grouped together in larger ones, or broken down in smaller ones, as well as the subsequent process of re-tagging the patents which were classified under the modified categories. This last process, in particular, may be extremely time-consuming and costly to implement.

APC systems can support the re-classification process by

- suggesting new (larger or smaller) categories, in particular through the use of clustering technologies (algorithms such as *K-Mean*, etc.);
- automatically re-tagging the patents according to the new patent categories.

The major issue in re-tagging patents is to build a training corpus, since there is no existing set of patents with the correct new categories to train the APC system. When the number of modified categories is not too important, the solution is generally to feed manually the new categories with typical examples; most often, 10 or 20 examples may be sufficient for the system to correctly identify the new category and automatically re-classify the rest of the patent collection.

### ***12.7.4 Prior Art Search***

APC systems are extremely useful to assist patent examiners in their prior art search. From a collection of several million patents, an algorithm like the *kNN* can retrieve,

in a matter of a few seconds, the ten patents which are closest to a submitted patent application. As underlined earlier, this tool is all the more interesting because it does not depend on classification codes: it browses the entire patent collection to find similar documents.

The IPCCAT application mentioned above, which is hosted on WIPO's website, provides for such a prior art search tool.

#### 12.7.4.1 Non-Patent Documents

It should be mentioned here that the classification and prior art search tools described above can be applied to all kinds of documents, not only patents. Technical literature, for instance, is a good playground: patent attorneys are eager to find the literature, which relates to a patent application, and a tool such as the  $k$ NN can be very efficient in this context.

Automated classification systems also have a bright future in the field of web mining: the search for novelty, for example, requires one to browse large volumes of documents on the Web. Classifying them automatically and finding the closest documents to a patent application may save considerable time and work.

## 12.8 Main Issues

### 12.8.1 Accuracy

Improving the accuracy of APC systems is the essential challenge today. Although these systems tend to be currently used as classification assistants, the ultimate goal for researchers in the field is to provide fully unsupervised systems, for instance to build pre-classification tools or to classify large volumes of patents in batch mode.

Several research tracks are being considered to improve the accuracy of APC systems.

- *More training data*: Adding training examples is one of the most immediate solutions. This process should essentially target the categories where samples are scarce. If no additional examples are available for a given category, it can be considered to make several copies of the available documents; this technique is called "oversampling". It allows at least a poorly represented category to "exist" in the system's classification space. Another approach is to find non-patent documents (such as technical literature) strictly related to the category's topic in order to add words, which are typical of that topic. The underlying problem here is not that patent collections, classified by human experts, are not available, but that they are not *readily* available, i.e. they are generally privately owned and not available on commercial terms, or at relatively high prices.

- *Better training data*: The patents provided as training examples must be accurately classified under the most recent version of the classification. They must also be recent: old patents may belong to categories, which no longer exist (if they were not re-classified) and thus may blur the information provided by other examples. Two techniques may be used in particular to improve the quality of the training set: the first is using a time-window which will be moved over time so as to keep only patents which were granted over the last, say, 15 years. Thus every year the latest patents are added, the oldest ones are removed, and the system is re-trained. The second one is to use a so-called “validity file”: this file defines all the categories, which are valid in the latest version of the classification. Before using a training set, all the categories assigned to its patent examples are compared to the validity file, and the examples whose categories are no longer valid are removed. This eliminates noise in the training set.
- *Building a “Committee of Experts”*: It was described above that not all sections of a patent are used to train the classifier. The situation is in fact somewhat more complex: some categories (such as, for instance, chemistry or electronics) may be best described by specific sections (such as the title or the inventor and applicant names), while other categories could be better characterized by other sections (such as the abstract). This is partly because some inventors or applicants are very specialized, and some fields use very specific words while others (e.g. for more general or conceptual inventions) are described with a broader vocabulary and thus need more information to be specified. One possible solution is to build a so-called “Committee of Experts”: one technology (for example neural networks) is used to create a large number of classifiers, each of them being built on different patent sections and being tested against all the classification categories. Then another technology (in this example, it would be an SVM machine) is used to assess which classifier is more fit to which category. Let us imagine that the best classifier for a given category of electronic devices is the one based on the inventor’s and applicant’s names, while for some agricultural devices it is the one using the abstract and first 350 words of the full text description. When a patent application is submitted, all the neural networks will be required to make predictions, but the SVM machine will favor the answer of the one which it found best fit to the specific context of that application.
- *Using linguistic processing*: In order to gain classification accuracy, the general need is to add information to help the APC system to draw clearer separations between each category.

Linguistics can help. An important step may be to disambiguate the vocabulary by using so-called *word sense disambiguation*. This may be done by using the context of the word in combination with a semantic network (a so-called *ontology*) to help the system discriminate between several possible senses of a word.

The use of *collocations* and *compound words* also help to discriminate between concepts. A special program looks at the co-occurrence of all the words in all the training examples, and if some words occur together very frequently they are automatically considered a single entity. In the example given above, “spark plug”

would be processed as one term. In recent experiments, using collocations allowed an NN-based APC to improve its precision score by up to 5%.

If there are too many words, an efficient solution to reduce the number of words is to use *stemming*: only the radical part of the word is index, which allows one to group several words with a common beginning but different endings due to plural or feminine forms, conjugated verbs, etc. The efficiency of stemming depends on the algorithm chosen: in the case of neural networks, stemming actually proved counter-productive because it tended to blur the concepts behind the words.

Another technique called *n-gram processing* is useful for some languages with specific issues.

- German, for example, is an agglutinative language, i.e. it can stick together several words when they are used together. This creates a “new word” for the indexer, so it may be important to find back the compounding words in order to limit the size of the index and get more examples of the words considered.
- Chinese and other languages which use ideograms (symbols) instead of letters are difficult to segment into words: 2, 3, 4 or more ideograms can compose a word, so specific rules may be used to solve this issue.

*N*-gram processing is a technique by which the first *n* letters (2, 3, 4 or more) which are read by the indexer are built into a word, then the following 2, 3, 4 letters, starting from the second character, are built into another word, and so on. The system stops when a blank space is met. A 2-letter *n*-gram rule is called a bi-gram, a 3-letter rule is a 3-gram, etc. This technique allows us to keep the words which were found both independently, and within a larger string, thus enriching the information provided by the training examples.

All the techniques proposed above can be used in combination in order to maximize the chances to classify more accurately. For example, it may be a good idea to test various linguistic processing techniques and to add the best-performing ones when building a Committee of Experts.

### ***12.8.2 Scalability***

Most APC tools currently classify patents down to Main Group level with a reasonable accuracy level. A common request from users is now to classify at Sub-group level, which means an increase in the number of possible categories by about a factor 10. It also implies to work with much larger training sets since examples are needed for each individual category.

A tool such as the *k*NN has intrinsic scalability issues, which tend to be more often studied in the field of search engines because its scalability depends on the size of the corpus (not of the classification). As mentioned earlier, from the point of view of APC it is too slow because it has no training phase so it cannot be prepared in advance.



As for the scalability of SVM, as far as we know, this issue is not solved yet because what would be needed is a highly efficient and robust implementation of a parallelized architecture—which does not seem to have been implemented so far. Some interesting research work is being done in this field, in particular by implementing SVM on Graphical Processing Units (GPUs), which have hundreds of processors.

For systems based on neural networks, the size of the neural networks to be built is the product of the number of words and of categories. The problem is not linked to the calculation power because it is possible to use many processors simultaneously for the training phase. However, to be efficient, the neural networks must be stored in RAM memory. Thus the main limit to NN systems currently lies with the RAM capacity. For example, in order to build an APC system over 70,000 categories and 3 million words the system would need about 256 Gb of RAM. This type of architecture is in fact available but it is still costly.

## 12.9 Conclusion

Artificial intelligence is still largely perceived as a “magical” resource and expectations tend to be excessive with regards to its real capacities and to the services it can provide. It is not possible for AI systems to classify with regards to *any* sort of classification; the classification needs to make some sort of sense to the APC system, i.e. the system must be able to draw clear limits between the categories.

Additionally, classifying on words has intrinsic limits: it is not always possible to define or represent categories with words. For example, it would probably be very difficult to describe with patent examples the category of inventions, which are easy or difficult to implement, or the category of inventions, which are profitable, or not.

The good news is that, so far, most categories of the IPC or ECLA are well represented and recognized, with the notable exception of the chemistry field, for which specific tools making use of graphics have now been developed.

Clearly today, the main challenge is to improve the accuracy of APC systems at the lowest levels of patent classifications. This will essentially be achieved by adding information, but not just *any* information because this can be counterproductive. Another promising approach will be to specialize the APC systems according to the various patent fields, for instance by choosing the most appropriate technologies for each particular family of topics. Besides, in addition to more powerful technologies and equipment, APC systems will also require larger and better data sets to be trained, tested and improved.

When all these conditions are gathered, APC applications can indeed become very effective and useful tools, both as assistants to human experts and as independent tools, and both for pure organizational tasks and for information retrieval purposes. For that reason their use is most probably going to expand fast in the future.

## References

1. Sebastiani F (2002) Machine learning in automated text categorization. *ACM Comput Surv* 34(1):1–47

## *Further Reading*

2. WIPO's website page dedicated to international patent classifications (IPC, Nice, Locarno, Vienna): <http://www.wipo.int/classifications/en/>. Accessed 23 Dec 2010
3. EPO's website page dedicated to ECLA: <http://test.espacenet.com/ep/en/helpv3/ecla.html>. Accessed 23 Dec 2010
4. World Patent Information, Elsevier: an International Journal for Industrial Property Documentation, Information, Classification and Statistics (Quarterly)
5. Berry MW, Castellanos M (eds) (2007) Survey of text mining: clustering, classification, and retrieval. Springer, Berlin
6. Fall CJ, Töröcsvári A, Benzineb K, Karetka G (2003) Automated categorization in the international patent classification. *SIGIR Forum* 37(1)
7. Fall CJ, Benzineb K, Guyot J, Töröcsvári A, Fiévet P (2003) Computer-assisted categorization of patent documents in the international patent classification. In: Proceedings of the international chemical information conference (ICIC'03), Nîmes, France, Oct 2003
8. Proceedings of the CLEF-IP 2010 (classification task), to be published in 2011. The related web site is here: <http://www.ir-facility.org/research/evaluation/clef-ip-10>. Accessed 23 Dec 2010

# Chapter 13

## Phrase-Based Document Categorization

Cornelis H.A. Koster, Jean G. Beney, Suzan Verberne, and Merijn Vogel

**Abstract** This chapter takes a fresh look at an old idea in Information Retrieval: the use of linguistically extracted phrases as terms in the automatic categorization of documents, and in particular the pre-classification of patent applications. In Information Retrieval, until now there was found little or no evidence that document categorization benefits from the application of linguistic techniques. Classification algorithms using the most cleverly designed linguistic representations typically did not perform better than those using simply the bag-of-words representation. We have investigated the use of dependency triples as terms in document categorization, according to a dependency model based on the notion of aboutness and using normalizing transformations to enhance recall. We describe a number of large-scale experiments with different document representations, test collections and even languages, presenting evidence that adding such triples to the words in a bag-of-terms document representation may lead to a statistically significant increase in the accuracy of document categorization.

### 13.1 Introduction

The document representation most widely used today in Information Retrieval (IR) is still the *bag-of-words* model, both in traditional query-based retrieval and in automatic document categorization.<sup>1</sup> In this document representation, the order of the

---

<sup>1</sup>The terms Categorization and Classification are used interchangeably.

C.H.A. Koster (✉) · S. Verberne · M. Vogel  
Computing Science Institute ICIS, Univ. of Nijmegen, Nijmegen, The Netherlands  
e-mail: [kees@cs.ru.nl](mailto:kees@cs.ru.nl)

S. Verberne  
e-mail: [s.verberne@cs.ru.nl](mailto:s.verberne@cs.ru.nl)

M. Vogel  
e-mail: [merijnv@cs.ru.nl](mailto:merijnv@cs.ru.nl)

J.G. Beney  
Dept. Informatique, LCI, INSA de Lyon, Lyon, France  
e-mail: [jean.beney@insa-lyon.fr](mailto:jean.beney@insa-lyon.fr)

words in a document plays no role at all, whereas our intuition tells us that important information about the category of a document must be available in certain groups of words found in the document. It is as if we are trying to determine the function of a ruined building (church? castle? monastery? dwelling? shop? railway station?) by dismantling it into bricks and then looking exclusively at the frequency distribution of the individual types of bricks, by disregarding any coherent substructures but simply taking them apart into individual bricks (the bag-of-bricks model). Intuitively it seems obvious that larger units (combinations of words into meaningful phrases) could serve as better terms for the categorization, yet there is surprisingly little evidence corroborating this intuition.

In this paper we describe a successful attempt to improve the accuracy of classification algorithms for patent documents by using, besides words, dependency triples (see Sect. 13.2) as features in the document representation.

### 13.1.1 Previous Work

The use of linguistically derived phrases as indexing terms has a long history in IR (for an overview see [26]). Many different kinds of linguistic phrases have been tried, with at most a modest success [4]. The predominant feeling about the value of NLP to IR, as voiced in [26], is that only ‘shallow’ linguistic techniques like the use of stop lists and lemmatization are of any use to IR; the rest is a question of using the right statistical techniques.

In the literature on document classification two forms of linguistic phrases are distinguished:

- *statistical phrases*, chunks or collocations: sequences of  $k$  non-stop words occurring consecutively [7], stemmed and ordered bigrams of words [6], even collocations taken from a specially prepared domain terminology [3];
- *syntactic phrases*, identified by shallow parsing [1, 18], template matching, finite state techniques [11] or by “deep” parsing [10, 14, 23, 27].

In spite of all these efforts, over the years no classification experiment using syntactic phrases has shown a marked improvement over the use of single keywords, at least for English. Only [23] recently reported a positive effect in classifying the Reuters-21578 collection. In the case of statistical phrases, the experience is only a little bit more positive [6, 18].

In this paper, we show that the use of syntactic phrases can significantly improve the accuracy of patent classification when each document is represented by a bag of words and dependency triples, generated according to a new aboutness-based dependency model, using deep parsing and transduction.

In the remainder of this section we describe the IR notion of aboutness. We propose an indirect way to measure aboutness and formulate the aboutness hypothesis. In section two of this chapter we will describe the use of dependency triples as terms for IR, introducing and motivating a dependency model based on the notion

of aboutness and discussing the syntactic normalization of phrases, which is crucial for achieving recall.

In section three we describe the software and data resources used in our experiments: the test corpora, parsers and classification engines used. In section four and five we describe our experiments to test the effect of using dependency triples according to this model in the classification of patent documents, investigating the plausibility of the aboutness hypothesis and presenting the experimental results. Finally, we formulate our conclusions.

### 13.1.2 *The Notion of Aboutness*

The notion of *aboutness* is highly central to Information Retrieval: the user of a retrieval system expects the system, in response to a query, to supply a list of documents which are *about* that query.

A model-theoretic basis for the notion of aboutness was described in [5]:

An information carrier  $i$  will be said to be *about* information carrier  $j$  if the information borne by  $j$  holds in  $i$

The rather abstract notion of “information carrier” can denote a single term, but also a composition of terms into a structured query or a document. Practical retrieval systems using words as terms are in fact based on a surprisingly simple-minded notion of aboutness:

If the word  $x$  occurs in the document then the document is *about*  $x$ .

This notion may appear highly naive, but it leads directly to the vector-space model when a measure for the *similarity* between the query and the document is introduced, based on the aboutness of words, the term frequency and document frequency.

For phrase-based retrieval, we need a notion of aboutness appropriate for phrases, which in its simplest form can be defined in the same way:

If the phrase  $x$  occurs in the document then the document is *about*  $x$ .

But the problem with all these definitions is that they are not concrete enough to compare the aboutness of different document representations.

Although intuitively it seems obvious that linguistic phrases provide a more informative document representation than keywords, the above formulation is not helpful in deciding what phrases from a document to choose and how to represent them.

### 13.1.3 *Measuring Aboutness*

In fact we cannot measure aboutness directly for lack of a well-defined measure, but we can compare the accuracy (precision and recall) achieved in the categorization

of a given set of documents using different document representations: *the representation that leads to the highest accuracy must best capture the aboutness of the documents.*

Text Categorization provides a good vehicle for the study of document representations, because many suitable test sets are available and accuracy can be measured objectively. Categorization does not suffer from the problem of measuring recall, unlike query-based search. Classification algorithms with a firm statistical basis allow objective and repeatable experiments, which can easily be replicated by others.

Successful classification algorithms are purely based on the frequency distribution of the *content words* (non stop-words) occurring in the different categories (forming the “language models”), and therefore on aboutness, instead of being based on any deep semantic analysis and reasoning. Statistics is a good and cheap alternative for the Semantics which is as yet unattainable.

### 13.1.4 The Aboutness Hypothesis

According to Information Retrieval folklore, the best classification terms are the *content words*: the words from the open categories, in particular nouns and adjectives. The words from the closed categories are just stop words. In a classification experiment reported in [2], it was indeed found that nouns, verbs and to a lesser degree adjectives and adverbs are the only lexical words that contribute measurably to the aboutness of a text.

These observations lead us to the following *aboutness hypothesis*:

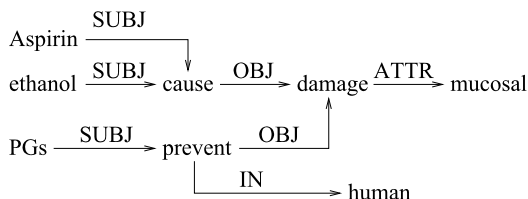
- of all the words in a text, the words from the open categories (nouns, verbs, adjectives, adverbs) carry most of the aboutness;
- of all possible dependency triples (words from the text joined by a syntactic relator), the triples containing only words from the open categories will carry most of the aboutness;
- we expect a better classification result when using triples as terms in addition to words than when using words alone.

We are going to investigate the elements of this hypothesis in one of the hardest document classification tasks: the classification of patent documents.

## 13.2 Dependency Graphs and Dependency Triples

By a *dependency graph* for a phrase we mean an acyclic graph (a tree with possibly additional confluent arcs) whose nodes are marked with words from that phrase and whose arcs are marked with directed relations. We are particularly interested in *syntactic relations*, those that can reliably be found by parsing.

As an example, we assign the following dependency graph to the sentence ‘In humans, PGs prevent the mucosal damage caused by aspirin and ethanol’:



A dependency graph represents the main structure of a sentence in an abstract way, much more compactly than a constituent tree (parse tree), in terms of certain syntactic relations between words of the sentence. It represents the compositional structure of the sentence, over which semantic relations can be constructed relatively easily. In that sense it is close to a semantic representation.

By a *dependency triple* we mean a triple [word, relation, word], which forms part of a dependency graph, from which it can be obtained by *unnesting* the graph to the triples contained in it. Triples are a syntactic way of representing phrases, closely related to the Head/Modifier pairs often used in IR (e.g. [10]).

### 13.2.1 The Aboutness-Based Dependency Model

The following table shows an example triple for each of the most important dependency relations:

subject relation	[device, SUBJ, comprise]
object relation	[cause, OBJ, lesion]
predicate relation	[aspirin, PRED, NSAID]
attribute relation (adj.)	[damage, ATTR, mucosal]
attribute relation (noun)	[breaker, ATTR, crust]
prepos relation (noun)	[coating, PREPof, albumin]
prepos relation (verb)	[charge, PREPinto, container]
prepos relation (adj.)	[related, PREPto, serotonin]
modifier relation (verb)	[extending, MOD, angularly]
modifier relation (adj.)	[efficient, MOD, highly]

Notice that in each of those relations both the head and the modifier are content words. In this respect, the aboutness-based dependency model is quite different from traditional linguistically-motivated dependency models that describe dependency relations between *all* words in the sentence, including punctuation, such as Minipar [20] and Link Grammar [25]; it is closer to the collapsed version of the Stanford Dependency Parser [9] output. Furthermore, the *subject* is taken as the head of a clause rather than the verb. This has some advantages: it allows one head to be the subject of several verbal clauses, and makes it easy to express certain raising constructions.

### 13.2.2 *Factoids*

The above dependency relations express only the factual content of the sentence, without further frills like time and modality of the verbs, topicalization, the argumentation structure (as expressed by conjunctions) and the linear word order, which are lost in this graph representation. Aboutness-based dependency graphs are well-suited to represent *factoids*, simple sentences expressing a (purported) fact. A typical factoid pattern is ‘who did what to whom with what’.

From the above example sentence, the following factoids can be derived:

- Aspirin causes mucosal damage
- ethanol causes mucosal damage
- PG’s prevent [that] damage in humans.

### 13.2.3 *Normalization*

In writing a text, people avoid literal repetition: they will go out of their way to choose another word, another turn of phrase, using anaphora, synonymy, hypernymy, periphrasis, metaphor and hyperbole in order to avoid the literal repetition of something they have said or written before. From a literary standpoint this is wonderful, but it gives complications in IR: we have to compensate for this human penchant for variety.

Rather than using literal phrases taken from the text as terms, we therefore reduce each phrase to a normal form which expresses only the bare bones of its aboutness. We eliminate all dispensable ornaments and undo such morphological, syntactic and semantic variation as we can, while conserving the aboutness.

The *syntactic normalization* is in our case expressed in the grammar underlying our parser, using a mechanism for describing compositional transduction: every construct in the grammar is described along with its transduction to the output format (dependency graphs with content words as heads and modifiers).

- All elements which do not contribute to the aboutness of the text are elided during transduction: articles and determiners, quantifiers, auxiliary verbs, conjunctions—which is much like applying a stop list;
- embedded constructions such as relative clauses and participial constructions, like the one in the example sentence, are expanded into additional basic sentences (and therefore SUBJ and OBJ relations);
- a preposition linking two content words is treated as a parameter to the PREP relation, bringing the content words cw1 and cw2 together into one triple [cw1, PREPpreposition, cw2], rather than producing triples like

```
[content-word-1, PREP,preposition]
[preposition, PREP', content-word-2]
```

in which the preposition (a non-content word) occurs as a head or modifier;



- for the same reason, in triples involving a copula, the copula is elided: ‘*the car is red*’ leads to the triple [car, ATTR, red], just like ‘*a red car*’;
- one of the most effective normalizing transformations is *de-passivation*: transforming a passive sentence into an active sentence with the same aboutness. By this transformation, the sentence renal damage caused by Aspirin is considered equivalent to Aspirin causes renal damage.

*Morphological normalization* by lemmatization is applied to the nouns and main verbs occurring in the resulting triples. As an example, the triples

[model, SUBJ, stand]      [stand, PREP, window]

can be obtained from ‘the model stood at the window’, ‘a model standing at the window’ and from ‘two models were standing at the window’. However the phrase ‘a standing order’ leads to a triple [order, ATTR, standing] because in this case ‘standing’ is not a main verb.

By these normalizing transformations, we try to improve the recall while surrendering little or no precision.

### 13.2.4 Bag of Triples and Bag of Words

In any language, there are many more triples (pairs of words with a relator between them) than words. When these are used as terms in a bag-of-triples model, the classification algorithm must be able to cope with very high numbers of features, demanding sophisticated Term Selection. But for the same reason, the feature space is very sparse. Triples may be high-precision terms, but they threaten to have a very low recall: a triple is never more frequent than the lowest frequency of the words of which it is composed. Low-frequency triples abound.

In classifying documents on e.g. gastric ulcers, it is therefore not immediately clear whether a triple like [ulcer, ATTR, gastric] is a more accurate search term than the co-occurrence of the words ‘ulcer’ and ‘gastric’. It will probably have a higher precision but also certainly a lower recall. Compared to the literal string ‘gastric ulcer’ the triple will have the same precision but a slightly higher recall, since it catches also phrases like ‘most of the ulcers appeared gastric’. Does the gain in precision compensate for the loss in recall? We need experimentation in order to find out whether triples work.

## 13.3 Resources Used

In this section we describe the resources used in the experiments: the corpora, the classification system and the parser. The next section describes the experiments that were carried out and their results.

### 13.3.1 *The CLEF-IP Corpora*

The Intellectual Property Evaluation Campaign (CLEF-IP)<sup>2</sup> is an ongoing benchmarking activity on evaluating retrieval techniques in the patent domain. CLEF-IP has two main goals:

- to create a large test collection of multi-lingual European patents
- to evaluate retrieval techniques in the patent domain

In 2010, CLEF-IP uses a collection of some 2,000,000 patent documents with text in English, German, and French, part of which was used in our experiments. In the remainder, we shall designate this corpus by the name CLIP10. The total number of files in the CLIP10 corpus is 2,680,604, in three different languages. We classified only the abstracts, focusing on the two languages for which we have a parser, and taking only those documents for which an IPC-R classification is available. We used two sub-corpora:

- CLIP10-EN: abstracts taken from all files in English having one or more IPC-R classifications
- CLIP10-FR: the same for French

We considered only those classes and subclasses for which at least one document was available. Some statistics about these sub-corpora:

	CLIP10-EN	CLIP10-FR
nmb of documents	532,274	55,876
avg nmb of words per document	119.5	121.2
nmb of classes	121	118
nmb of subclasses	629	617
avg nmb of classes/doc	2.13	1.32
min/max nmb of classes/doc	1/14	1/7
avg nmb of subclasses/doc	2.72	1.42
min/max nmb of subclasses/doc	2/19	1/7

### 13.3.2 *The English Parser AEGIR*

The experiment on CLIP10-EN made use of a new hybrid dependency parser for technical English, which combines a rule-based core grammar (Accurate English Grammar for IR) and a large lexicon of technical terms with various disambiguation mechanisms. The experiments described here were its first large-scale test. AEGIR

<sup>2</sup>See [www.ir-facility.org/research/evaluation/clef-ip-10](http://www.ir-facility.org/research/evaluation/clef-ip-10).

is a further development of the EP4IR parser (English Phrases for Information Retrieval) [16]. In those experiments involving words, the words in the documents were case-normalized but not lemmatized; punctuation was removed.

The words occurring in the triples were case-normalized, and *cautiously* lemmatized: nouns were brought into their singular form, and main verbs brought into their infinitive form. When not used as main verb (e.g. “related”), participles were not lemmatized. Verb particles were attached to the verb lemma (e.g. “turn\_back”).

The parser was running on a relatively small parallel computer (four CPU’s with four cores each) with a very large main memory (128 Gigabytes).

### 13.3.3 The French Parser FR4IR

The French documents were parsed with the French parser FR4IR developed by Jean Beney at the INSA de Lyon. A description of the FR4IP parser can be found at [www.agfl.cs.ru.nl/FR4IR](http://www.agfl.cs.ru.nl/FR4IR) where there is also a demonstration version. The parser is not at present in the public domain, but for scientific purposes a copy of the parser can be obtained from [jean.beney@insa-lyon.fr](mailto:jean.beney@insa-lyon.fr). It is still very much under development.

The preprocessing of the French documents was similar to that for English. All experiments with French documents were performed on the Large Data Collider (LDC) of the Information Retrieval Facility ([www.ir-facility.org](http://www.ir-facility.org)).

### 13.3.4 The LCS Classification Engine

The Linguistic Classification System (LCS) used in these experiments [15] was developed at the University of Nijmegen during the IST projects DORO (Esprit 22716, 1997 to 1999) and PEKING (IST-2000-25338, 2000 to 2002) for the experimentation with different document representations. In the current TM4IP project<sup>3</sup> it was re-implemented in Java. For most of the experiments we used the Balanced Winnow algorithm [8, 21, 22], a child of the Perceptron, which is very efficient and highly robust against large but sparse feature sets. Some of the experiments were also performed using the Support Vector Machine algorithm (SVM light [12]).

As is the case for other classification engines, the performance of the classifiers depends on certain tuning parameters. In the experiments with Winnow in this section the parameters were chosen based on some small tuning experiments on training data:

Promotion parameters:  $\alpha = 1.02$ ,  $\beta = 0.98$

Thick threshold parameters:  $\theta^- = 0.5$ ,  $\theta^+ = 2.0$

Number of Winnow iterations = 10

---

<sup>3</sup>[www.phasar.cs.ru.nl](http://www.phasar.cs.ru.nl).

Global term selection:  $DF > 1$ ,  $TF > 2$

Local term selection: Simplified ChiSquare, max 10,000 terms/category.

For SVM, we used  $C = 2$ ; we only experimented with the linear kernel.

In combining triples and words, both were simply considered as literal terms (the two term lists were concatenated). We did not try to give them different weights.

Since patents may have multiple IPC-R labels, we perform a multiclassification. After training classifiers for each category, the test documents were given a score by each classifier and each test document that scored better than a certain threshold for a certain category was assigned to that category.

The Winnow algorithm has a natural threshold 1 for each category. Each test document was assigned to at least one category (the one with the highest score) and at most five categories for which its score was greater than 1. Micro-averaged Precision, Recall and F1 were then computed in the usual way by comparing these categories with the ones for which the document is labeled.

This procedure reflects a pre-classification situation: each incoming document must be sent to precisely the right examiners.

For other measures (Recall or Precision at  $k$  documents, etc.) for all categories a full ranking of all test documents is produced.

## 13.4 Experimental Results

In the following experiments we have investigated the aboutness hypothesis, using document classification:

- what is the relative contribution of the open word classes?
- which dependency relations contribute most to the aboutness?
- is classification on words plus triples more accurate than classification on words alone?

Our main experiment concerns the comparison of three document representations: words alone, triples alone and words plus triples. As the measure of accuracy we used the Micro-averaged F1, assuming a loss function attaching equal weight to precision and recall, which is representative for applications in patent pre-classification.

### 13.4.1 Experimental Setup

The experiments in this section were each performed 10 times (10-fold cross-evaluation). In each experiment on unseen documents the documents were split at random into a train set (80%) and a test set (20%). After training on all train and test sets, we determined the micro-averaged value and standard deviation of the precision, recall and F1.

For the experiments with seen data we trained on a random subset of 80% of the documents and used a quarter of these train documents as test set. As usual, the

terms ‘seen’ and ‘unseen’ refer to testing the accuracy of the classifier on the train set used and on the held-out test sets, respectively.

In the tables, a number in brackets indicates the standard deviation. The rightmost column shows for each representation the improvement with respect to the baseline.

We ran these experiments with both the Winnow and SVM classifiers, in order to compare the effect of triples for those algorithms. Winnow and SVM are both known to cope well with large numbers of terms (in this case triples).

## 13.4.2 Winnow—Results for English

### 13.4.2.1 Testing on Seen Data

Algorithm	P	R	F1	improvement
Testing on seen documents				
Winnow	CLIP10-EN/classes			
words alone	83.36 (0.08)	72.05 (0.23)	77.30 (0.10)	baseline
triples alone	88.61 (0.09)	73.82 (0.19)	80.54 (0.15)	+3.24
words+triples	89.65 (0.08)	80.41 (0.06)	84.78 (0.07)	+7.48
Winnow	CLIP10-EN/subclasses			
words alone	83.93 (0.04)	65.15 (0.21)	73.36 (0.13)	baseline
triples alone	90.80 (0.02)	69.51 (0.05)	78.74 (0.03)	+5.38
words+triples	91.64 (0.04)	76.48 (0.05)	83.38 (0.04)	+10.02

The accuracy (F1) on *seen* documents has in all cases improved over that of words alone, both for triples alone and for words plus triples. This improvement may be attributed to the fact that there are so many more different triples than different words:

	different terms	after term selection
words	115,604	74,025
triples	7,639,799	1,177,134
words+triples	8,671,497	1,526,353

Using so many terms, it becomes easier to construct a class profile which models the differences between the classes. But these good results on seen data are not important for practical applications, since they do not carry over to unseen data.

### 13.4.2.2 Testing on Unseen Data

On unseen data, the picture is indeed somewhat different.

Algorithm	P	R	F1	improvement
Testing on unseen documents				
Winnow	CLIP10-EN/classes			
words alone	76.33 (0.14)	66.35 (0.13)	70.99 (0.07)	baseline
triples alone	74.26 (0.08)	58.71 (0.09)	65.57 (0.07)	-5.42
words+triples	78.25 (0.07)	69.06 (0.09)	73.37 (0.06)	+2.37
Winnow	CLIP10-EN/subclasses			
words alone	72.78 (0.13)	55.96 (0.16)	63.27 (0.07)	baseline
triples alone	71.80 (0.09)	48.64 (0.11)	57.99 (0.09)	-5.28
words+triples	75.72 (0.10)	59.31 (0.13)	66.52 (0.08)	+3.25

The results on *unseen* data are representative for the intended application (pre-classification of patent applications). The base line (words) may look disappointing, but it is quite hard to improve upon it. Attempts using lemmatization, stop lists etc. (not reported here) have consistently given no significant improvement.

As is to be expected, the classification on the 121 main classes is (by about 7 points) better than that on 629 subclasses. But in both cases the difference (of 2.37 or 3.25 percent points) between the base line of words and the triples plus words is many times the standard deviations. Therefore the difference is statistically highly significant—this is our main result.

Note that an accuracy below the base line is achieved when (unwisely) using only triples as terms. Both precision and recall are much lower for triples-only than for words, and so is the F1. Experts [10, 18] agree that linguistic terms should be *added* to the word terms, not used instead. Still, there is some progress: in our previous experiments [14] with Head/Modifier pairs we found the accuracy using only pairs to be much lower; probably the quality of the linguistic document representation and the parser has in the mean time improved, but not enough to “beat” the baseline (but see Sect. 13.4.5).

### 13.4.3 SVM—Results for English

For performance reasons, we ran the experiments with SVM on classes with four-fold instead of ten-fold cross-evaluation, and the experiment on subclasses only once, without cross-evaluation (one experiment classifying subclasses using words

and triples took 22 hours with SVM, compared to 2 hours with Winnow), but since the standard deviations appear to be quite low, we felt that performing a ten-fold cross-evaluation was not necessary.

Algorithm	P	R	F1	improvement
Testing on seen documents				
SVM	CLIP10-EN/classes			
words alone	90.32 (0.07)	72.31 (0.10)	80.32 (0.09)	baseline
triples alone	94.80 (0.07)	84.73 (0.04)	89.48 (0.05)	+9.16
words+triples	94.93 (0.03)	77.71 (0.05)	85.46 (0.04)	+5.14
SVM	CLIP10-EN/subclasses			
words alone	92.30	68.84	78.86	baseline
triples alone	96.66	87.75	91.9	+13.04
words+triples	98.71	91.23	94.82	+15.96
Testing on unseen documents				
SVM	CLIP10-EN/classes			
words alone	81.09 (0.12)	62.23 (0.14)	70.42 (0.14)	base line
triples alone	78.34 (0.09)	58.35 (0.11)	66.88 (0.10)	-3.53
words+triples	84.54 (0.01)	63.05 (0.04)	72.23 (0.02)	+1.81
SVM	CLIP10-EN/subclasses			
words alone	78.22	52.88	63.10	base line
triples alone	77.86	51.17	61.76	-1.34
words+triples	82.29	59.18	68.84	+5.74

The improvement in accuracy on seen documents was even larger for SVM than for Winnow. On unseen documents, a similar improvement when adding triples was found as was the case for Winnow, somewhat smaller for classes but larger for subclasses.

#### 13.4.4 Results for French

Again we trained on 80% of the documents and tested on 20%, with 10-fold cross-evaluation. We did not measure the accuracy on seen documents and used only Winnow. The results on *unseen* documents were as follows.

Algorithm	P	R	F1	improvement
Testing on unseen documents				
Winnow	CLIP10-FR/classes			
words alone	69.94 (0.21)	66.95 (0.24)	68.41 (0.19)	baseline
triples alone	54.41 (0.23)	46.28 (0.30)	50.02 (0.26)	-18.40
words+triples	72.84 (0.23)	67.50 (0.22)	70.07 (0.20)	+1.66
Winnow	CLIP10-FR/subclasses			
words alone	65.83 (0.20)	55.47 (0.22)	60.21 (0.18)	baseline
triples alone	46.45 (0.47)	35.74 (0.32)	40.27 (0.27)	-19.96
words+triples	68.67 (0.30)	56.01 (0.30)	61.70 (0.28)	+1.49

The baseline (words) is much lower on these French documents than for the CLIP10-EN documents because there are much fewer train documents (see also following section). The results for triples-only are quite bad, but again the addition of triples to the words does improve the classification result.

### 13.4.5 The Number of Train Documents

In this section we investigate the effect of the number of documents trained on the accuracy obtained. The following graph is a *learning curve*, obtained by testing on a 10% test set and training on increasing numbers of train documents, for the three different document representations and for two different measures of accuracy. The experiment was performed using Winnow on CLIP10-EN (classes) with 10-fold cross-validation.

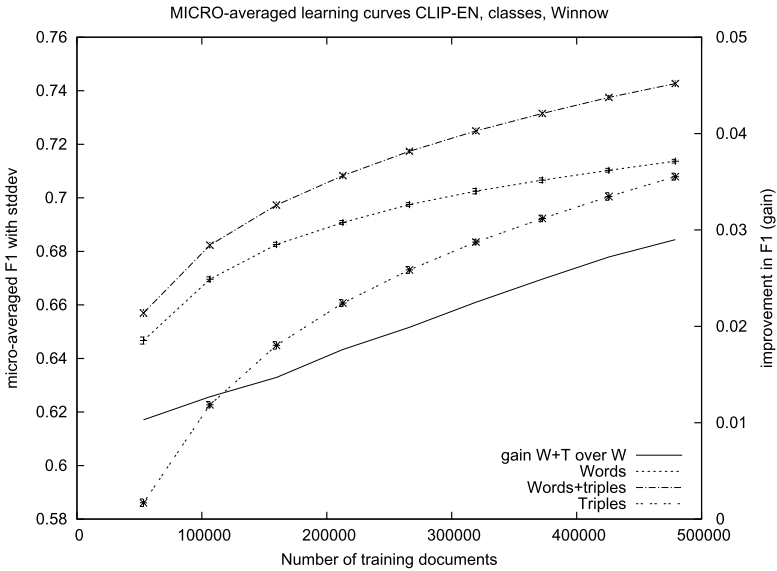
#### 13.4.5.1 The Effect for the Large Classes

In the first experiment, as in the experiments reported earlier, the *micro*-averaged F1 is reported (averaging over all <document, class> pairs).

Adding triples to words always gives an improvement and the improvement grows with the number of documents trained. Initially, the classification power of triples alone is lower than that of words, but it grows faster. It appears that, with



100,000 more train documents, the triples alone could achieve a better classification than the words alone, but we could not verify this.

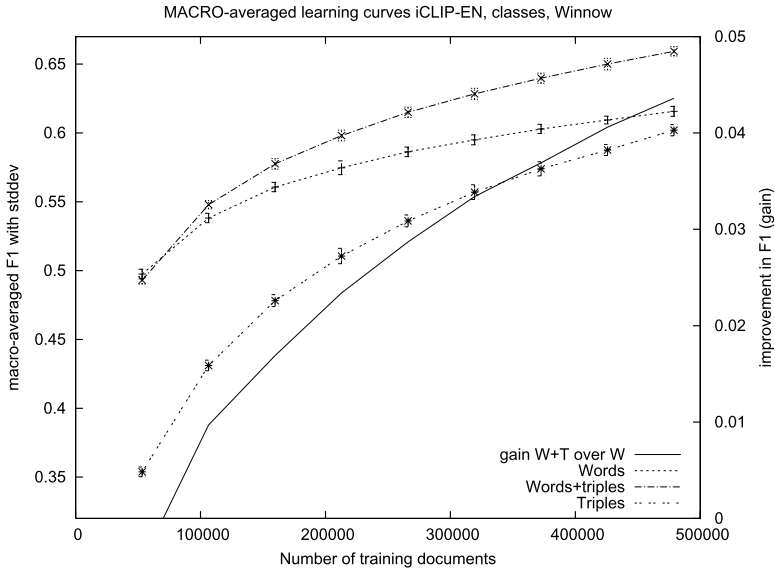


The improvement given by words + triples over words alone (the gain) grows about linearly with the number of train documents. This is good news for patent classification, where millions of train documents are available.

### 13.4.5.2 The Effect for Smaller Classes

The micro-average is dominated by the large classes, those with many documents. Therefore the previous graph does not show whether the effect for smaller classes is also positive. We therefore also show a graph of the *macro*-averaged accuracy (averaged over the classes) which assigns equal weight to all classes. In this experiment, 5-fold cross-validation was used.

The macro-averaged F1 is lower than the micro-averaged F1 (it is dominated by the smaller classes, which have (much) fewer train documents) but the gain in using triples is now larger (except for very small numbers of train documents). It appears from this experiment that triples may work even better for the smaller classes than for the large classes.



### 13.5 The Aboutness of Words and Triples

In this section we report on a number of experiments to shed light on the behavior of triples as classification terms. Because here we are more interested in insight than in representiveness, we use a smaller subset of CLIP10-EN, consisting of only the documents in class A61 of the IPC-R, a total of 73869 documents in 13 subclasses, varying in size from 208 to 38396 documents. We trained Winnow classifiers for its 13 subclasses on those documents (80% train, 20% test), and used the classification for the subclass A61B in those experiments in this section which concern a single class profile.

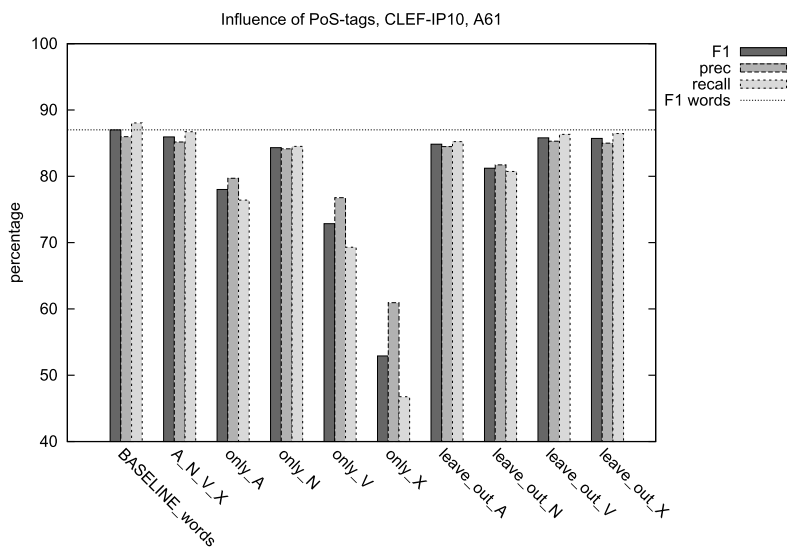
For these experiments with much fewer documents we used the default values for the Winnow parameters ( $\alpha = 1.1$ ,  $\beta = 0.9$ ,  $\theta^- = 0.5$ ,  $\theta^+ = 2.0$ , number of iterations = 3,  $DF > 1$ , no local term selection).

#### 13.5.1 The Aboutness of Word Categories

In [2], the contribution to the accuracy by one category of words was measured by classifying a given corpus using only words from that category as terms. The category of words was determined using the Brill tagger. Of course this tagging was not very precise, because the tagger takes only local context into account.

We have repeated this experiment on CLIP10-EN class A61, using the AEGIR parser as a tagger: during the parsing, the parser will disambiguate the Part-Of-Speech (PoS) of every ambiguous word in the sentence. Both the Brill tagger and the parser can (and will) make errors, but in our approach we can be sure that each occurrence of a word obtains a Part-of-Speech which is appropriate in its syntactic context.

The following graphs show the results of the classification of A61 (80/20 split, 10-fold cross-evaluation). In each case we measured the accuracy (F1) for the classification of the subclass A61B, either using *only* the words with a certain PoS or *excluding* them. We use the letters A, N, V and X to stand for the open word categories: adjective, noun, verb and adverb.



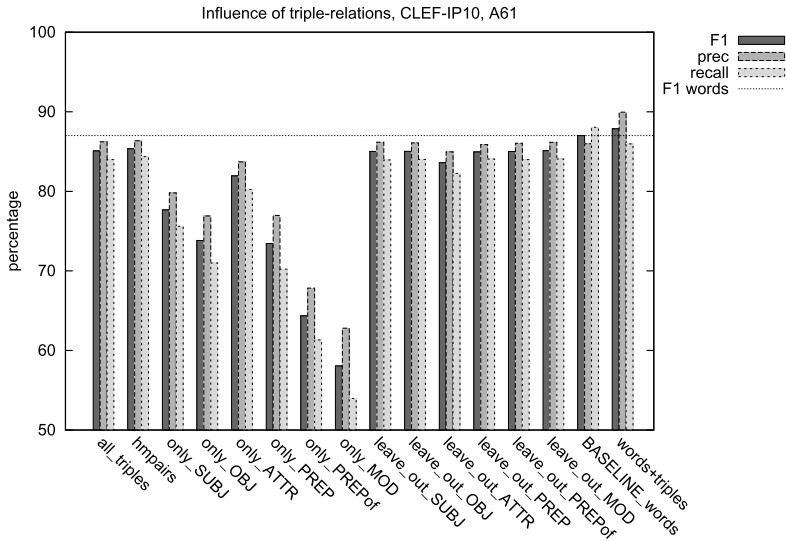
As was to be expected, the one-POS-only experiments show that the nouns have by far the highest aboutness, followed by the adjectives and the verbs. Adverbs have very little aboutness.

The leave-one-out results show less variation. The only word category that should not be left out are the nouns, the others make little or no difference. The accuracy when using all words is slightly better than when taking only those words that belong to one of the open categories (A+N+V+X), probably indicating some non-content words (e.g. numbers) that carry some aboutness.

### 13.5.2 The Aboutness of Relations

By a *relation* we here mean the set of all triples having a specific relator (e.g. ATTR or SUBJ). We want to measure for each relation what is its contribution to the accuracy of a classification.

We experimented again on the CLIP10-ENA61 corpus, this time representing each document by a bag of dependency triples, but either using *only* the triples of some relation or *leaving out* all triples belonging to that relation. The results for the subclass A61B are shown in the following graphs:



The ATTR relation by itself has the largest aboutness, because it represents the internal structure of the noun phrase and most of the terminologically important terms are noun phrases. Then follow the PREP, OBJ and SUBJ relations (which may be because they have a noun as head or modifier). The MOD relation, which has a verb as head, makes the smallest contribution to accuracy, but it is also the least frequent.

The leave-one-out experiments all give a very similar result. Only leaving out the ATTR relations leads to a marked drop in accuracy with respect to using all triples. Intuitively it is clear that the relations are not very independent terms, since the head of one triple is very often the modifier of another, or vice versa.

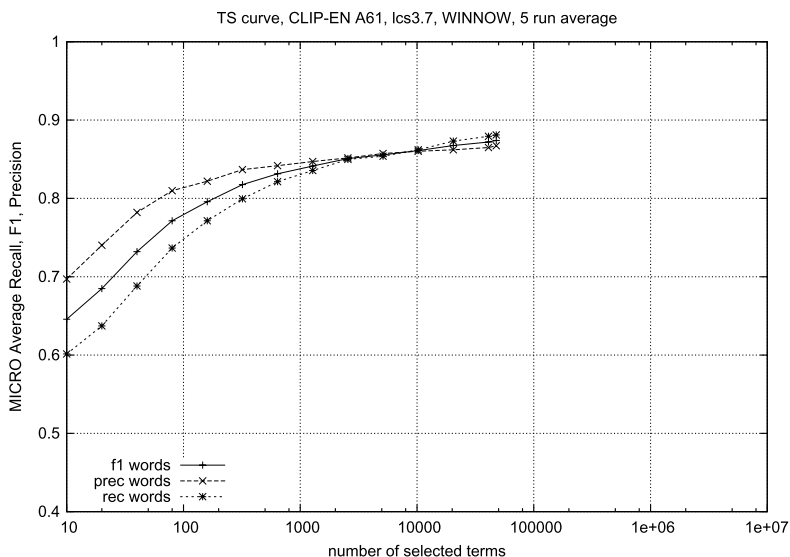
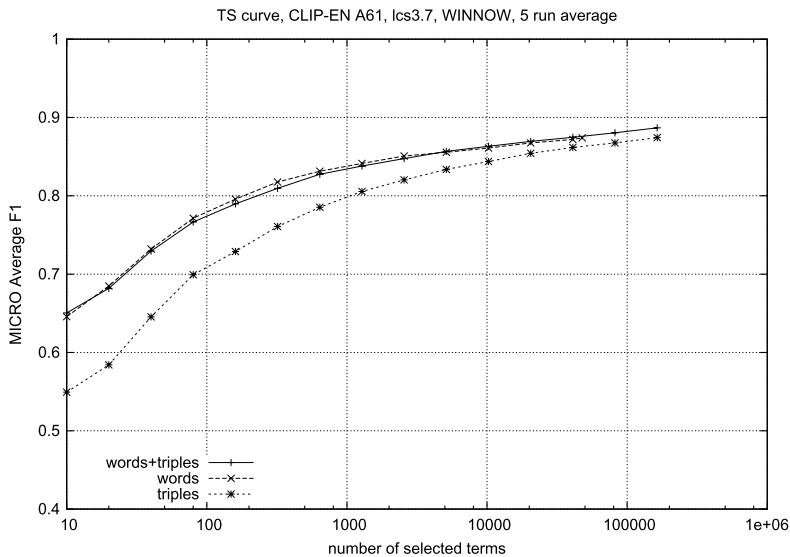
We also investigated the question whether the triple representation (with explicit relators) is better than the older Head/Modifier representation (obtained here by simply dropping the relator); the second graph from the left (marked hmpairs) shows a slightly higher recall and accuracy for Head/Modifier pairs.

### 13.5.3 Words and Triples as Terms

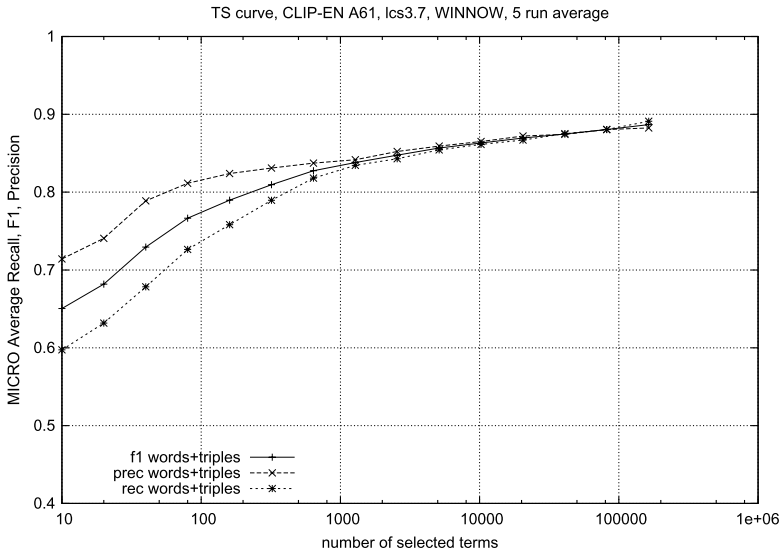
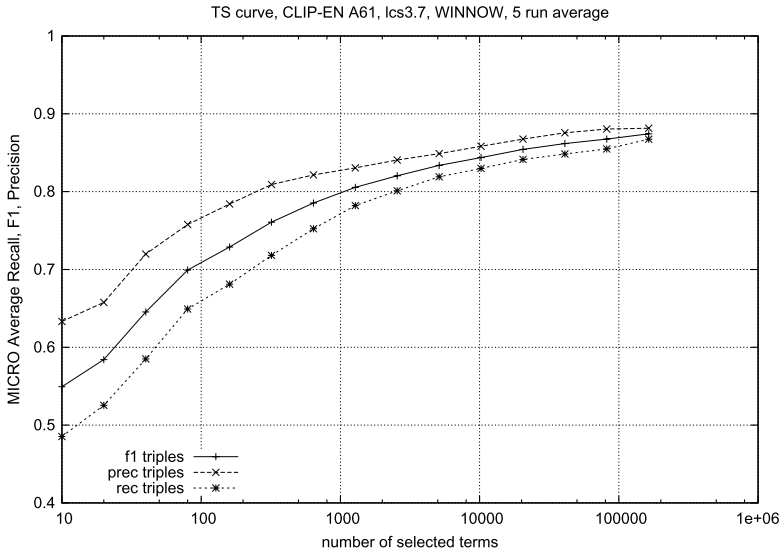
It appears that the use of triples as terms may lead to a better classification result, but many more of them are needed. Therefore we have compared the effectiveness

of words and triples in dependence on the *number* of terms used, in a Term Selection setting: we performed the same experiment (classifying A61 subclasses with Winnow with words alone, triples alone and words plus triples) while selecting for each subclass only the top  $n = 2^k$  terms using the simple Chi-square Term Selection (TS) criterion [24].

We show two graphs; the first is the resulting *TS curve* (F1 against number of terms) for the three representations, and the second shows in more detail the TS curve for words. The variance is so small that we did not show it in the graphs.



The TS curve shows that the triples do not start contributing to the accuracy (F1) until more than 10,000 terms are used; for fewer terms, using only words gives slightly better results. At the highest number of terms (beyond the end of the curve for words), triples plus words gives about 5 points improvement, while the F1 for triples alone equals that for words alone.



In the second graph, concerning words alone, we show also precision and recall. For low numbers of terms, precision exceeds recall, but between 2000 and 10,000

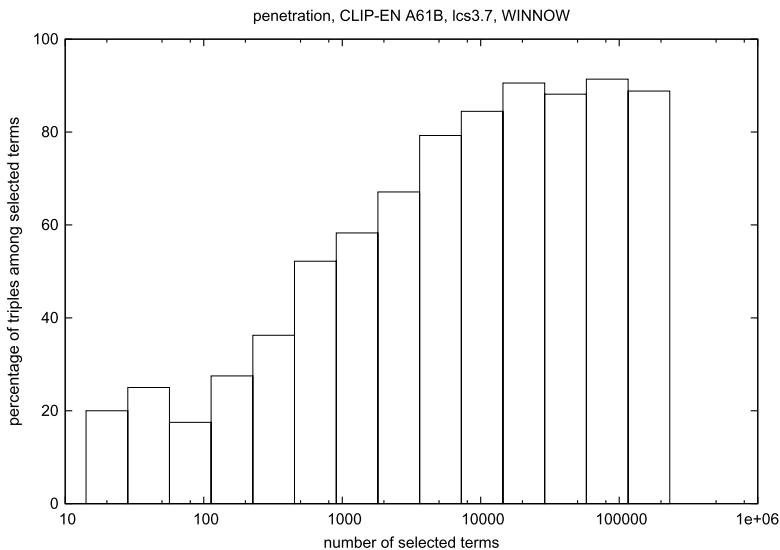
terms the two curves cross, and above 10,000 terms recall is larger than precision (but remember that this represents an average over 13 subclasses). Next, we look at the TS curves for the two other representations.

Using words plus triples, the cross-over point between precision and recall is now much higher than for words (between 40,000 and 100,000 terms). The graph shows that, when adding triples to words as terms, no term selection should be applied, because that may diminish the benefit.

For triples alone, precision is larger than recall in all cases, but it is possible (and looks likely) that a cross-over will occur for many more terms.

### 13.5.4 The Penetration of Triples

Another measure for the effectiveness of linguistic classification terms which has been used in literature [6] is the *penetration* of such terms in the class profile: in our case, the percentage of triples among the top  $n$  terms when classifying on both words and triples. We have measured it by classifying the subclasses of A61 (once) and then measuring the percentage of triples among the terms in the profile of A61B, in power of two buckets.



Among the 10 top terms are no triples, but after that more and more triples appear. After about 1000 terms the triples have a high penetration, but their presence does not improve the classification until very many triples are used. Inspection of the profile shows that, for fewer terms, the triples mostly replace words with more or less the same aboutness (as was found in [6]).

### 13.5.5 Discussion

The experiments described here were performed for the comparison of different document representations: words and words plus triples. For practical patent classification, of course many other considerations are important.

In classifying two different corpora of patent abstracts in two languages and with two different classification algorithms, a statistically significant increase in accuracy was found when adding triples to the word features (“bag of words and triples”). Further experiments are needed to see whether the same improvements are found when classifying other sections of the patents (e.g. the Claims) or non-patent documents.

In a previous article [13], we have described a similar series of experiments on patent abstracts and also on full-text documents, in which the results were very similar. For those experiments we used the EP4IR parser and LCS on the EPO2 corpus [17]. The presently used CLEF-IP corpus has the advantage of being publicly available, which makes it easier for others to duplicate our experiments.

It is interesting to note that all parsers used in the experiment are still under development. It appears likely that when the parsers reach more accuracy on the very complicated documents from the patent world, the effect of using triples may increase further.

The abundance of different triples each having only a low frequency still appears to be the main reason why the classification on triples alone gives disappointing results. We are now investigating whether this problem can be overcome by using some form of term clustering along the lines of [19].

The results on the aboutness of word categories and relations require further analysis, but it is definitely plausible that the aboutness hypothesis is well founded. Certainly not busted, we claim.

## 13.6 Conclusions

We have formulated the aboutness hypothesis, stating essentially that the open word categories are the carriers of aboutness, and introduced an aboutness-based dependency model, which differs from the more descriptive dependency models preferred in linguistics in that the heads and modifiers of the dependency graph involve only words from the open categories.

Dependency triples following this aboutness model are derived by syntactic parsing of the documents, followed by a transduction to dependency graphs and unnesting of the graphs to triples.

We have performed an experimental investigation on patent abstracts in English and French from CLEF-IP 2010. Our experiments showed that using aboutness-based dependency triples as additional features in document classification did lead to a statistically significant increase in classification accuracy, for larger as well for smaller classes. This improvement was found for different document collections in



different languages and for different classification algorithms. The larger the number of train examples, the larger the improvement was found to be.

In a classification setting, we have compared the aboutness of different categories of words and dependency triples. We also compared the behavior of words and triples using term selection and measured their penetration. All these experiments yielded rich evidence for the aboutness hypothesis, but this evidence still has to be analyzed theoretically and explained more deeply.

The two parsers and the LCS classification system used in the experiments are available from the authors for research purposes.

## References

1. Alonso M, Vilares J, Darriba V (2002) On the usefulness of extracting syntactic dependencies for text indexing. In: LNCS, vol 2464. Springer, Berlin, pp 3–11
2. Arampatzis A, Van der Weide T, Koster CHA, Van Bommel P (2000) An evaluation of linguistically-motivated indexing schemes. In: Proceedings of BCS-IRSG 2000 colloquium on IR research, 5th–7th April 2000, Sidney Sussex College, Cambridge, England
3. Bel N, Koster CHA, Villegas M (2003) Cross-lingual text categorization. In: Proceedings ECDL 2003. LNCS, vol 2769. Springer, Berlin, pp 126–139
4. Brants T, Google Inc (2003) Natural language processing in information retrieval. In: Proceedings CLIN 2003, pp 1–13
5. Bruza P, Huibers T (1994) Investigating aboutness axioms using information fields. In: Proceedings SIGIR 94, pp 112–121
6. Caropreso M, Matwin S, Sebastiani F (2000) A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In: Text databases and document management: theory and practice, pp 78–102
7. Cohen W, Singer Y (1996) Context sensitive learning methods for text categorization. In: Proceedings of the 19th annual international ACM conference on research and development in information retrieval, pp 307–315
8. Dagan I, Karov Y, Roth D (1997) Mistake-driven learning in text categorization. In: Proceedings of the second conference on empirical methods in NLP, pp 55–63
9. De Marneffe MC, Manning CD (2008) The Stanford typed dependencies representation. In: Coling 2008: Proceedings of the workshop on cross-framework and cross-domain parser evaluation. Association for Computational Linguistics, pp 1–8
10. Fagan J (1988) Experiments in automatic phrase indexing for document retrieval: a comparison of syntactic and non-syntactic methods. PhD thesis, Cornell University
11. Grefenstette G (1996) Light parsing as finite state filtering. In: Workshop on extended finite state models of language, ECAI'96, Budapest, Hungary, August 1996
12. Joachims T (1998) Text categorization with support vector machines: learning with many relevant features. In: European conference on machine learning (ECML-98). Springer, Berlin, pp 137–142
13. Koster CHA, Beney JG (2009) Phrase-based document categorization revisited. In: Proceedings of the 2nd international workshop on patent information retrieval (PAIR 2009) at CIKM, pp 49–55
14. Koster CHA, Seutter M (2003) Taming wild phrases. In: Proceedings 25th European conference on IR research (ECIR 2003). LNCS, vol 2633. Springer, Berlin, pp 161–176
15. Koster CHA, Seutter M, Beney JG (2003) Multi-classification of patent applications with winnow. In: Proceedings PSI 2003. LNCS, vol 2890. Springer, Berlin, pp 545–554
16. Koster CHA, Seutter M, Seibert O (2007) Parsing the medline corpus. In: Proceedings RANLP 2007, pp 325–329

17. Krier M, Zaccà F (2002) Automatic categorization applications at the European patent office. *World Pat Inf* 24:187–196
18. Lewis D (1992) An evaluation of phrasal and clustered representations on a text categorization task. In: *Proceedings ACM SIGIR'92*
19. Lewis D, Croft B (1990) Term clustering of syntactic phrases. In: *Proceedings SIGIR'90*, pp 385–404
20. Lin D (1998) Dependency-based evaluation of MINIPAR. In: *Workshop on the evaluation of parsing systems, Granada, Spain*
21. Littlestone N (1988) Learning quickly when irrelevant attributes abound: a new linear-threshold algorithm. *Mach Learn* 2:285–318
22. Littlestone N (2006) Generating typed dependency parses from phrase structure parses. In: *Proceedings LREC 2006*
23. Nastase V, Sayyad Shirabad J, Caropreso MF (2007) Using dependency relations for text classification. University of Ottawa SITE Technical Report TR-2007-12, 13 pages
24. Sebastiani F (2002) Machine learning in automated text categorization. *ACM Comput Surv* 34(1):1–47
25. Sleator DD, Temperley D Parsing English with a link grammar. In: *Third international workshop on parsing technologies*
26. Spairck Jones K (1999) The role of NLP in text retrieval. In: Strzalkowski T (ed) *Natural language information retrieval*. Kluwer, Dordrecht, pp 1–24
27. Strzalkowski T (1995) Natural language information retrieval. *Inf Process Manag* 31(3):397–417

# Chapter 14

## Using Classification Code Hierarchies for Patent Prior Art Searches

Christopher G. Harris, Robert Arens, and Padmini Srinivasan

**Abstract** Searches in patent collections to determine if a given patent application has related prior art patents is non-trivial and often requires extensive manpower. When time is constrained, an automatically generated, ranked list of prior art patents associated with a given patent application decreases search costs and improves search efficiency. One may view the discovery of this prior art patent set as a problem of finding patents ‘related’ to the patent application. To accomplish this, we examine whether semantic relations between patent classification codes can aid in the recognition of related prior art patents. We explore similarity measures for hierarchically ordered patent classes and subclasses for this purpose. Next, we examine various patent feature-weighting schemes to achieve the best similarities between our patent applications and related prior art patents. Finally, we provide a method and demonstrate that patent prior art searches can successfully be used as an aid in patent ranking.

### 14.1 Introduction

#### 14.1.1 Patent Searches

One of the primary responsibilities of a patent examiner is to perform a patentability test to check for novelty and non-obviousness/innovativeness of patent applications. To perform this task, a patent application is examined against a list of all

---

C.G. Harris (✉)

Informatics Program, The University of Iowa, Iowa City, IA, USA

e-mail: [christopher-harris@uiowa.edu](mailto:christopher-harris@uiowa.edu)

R. Arens

Nuance Communications, Burlington, MA, USA

e-mail: [robert.arens@nuance.com](mailto:robert.arens@nuance.com)

P. Srinivasan

Computer Science Department and Informatics Program, The University of Iowa, Iowa City, IA, USA

e-mail: [padmini-srinivasan@uiowa.edu](mailto:padmini-srinivasan@uiowa.edu)

patents with an earlier priority date, called prior art patents, for the possibility that the claims on a target and prior art patent overlap.<sup>1</sup> If the overlap is significant, it can lead to the rejection of the given patent application. Patent applications are evaluated using knowledge in any previous publication (such as in a published journal article) predating the patent application's priority date as described in Chap. 1 of this book; however, in this research where the purpose is to explore the value of patent hierarchies, we will limit our investigation of prior art to patents. Two types of patent searches are normally performed in patent examinations: 'novelty' or 'non-obviousness/inventiveness' searches, which help determine if an invention is novel prior to investing the time and effort to apply for a patent and 'clearance' or 'validity searches', which are done after a patent is issued to determine if a patent application can be invalidated by any prior patent or other work. We focus on the former of these two in this chapter.

Patents are classified into two different patent types: design and utility. Both types use the same classification systems. The key difference between design and utility patents is that a design patent protects "the ornamental design, configuration, improved decorative appearance, or shape" of an invention [32]. This patent is appropriate when the basic product already exists in the marketplace and is not being improved upon in function but only in style. A utility patent protects any new invention or functional improvements on existing inventions. This can be to a product, a machine, a process, or even a composition of matter. In the experiments we have conducted here, 97% of our dataset involves utility patents and 3% involve design patents.

Often patent prior art searches involve obtaining a list of patents and other publications and then manually refining the list, a process that is both laborious and prone to errors of omission. Moreover, the resources available for patent searches are frequently constrained by limitations of time or manpower; hence the need for an automatically generated ranked list of patents most likely to discover prior art for a given patent application. Therefore, our goal is to investigate methods that can provide an investigator with an ordered list of prior art patents identified from a patent collection. All of our methods here involve classification hierarchies. In particular we examine four hypotheses that explore the value of several different aspects of these classification hierarchies, which are explained further in Sect. 14.3.

### ***14.1.2 Using Classification Codes versus Keywords in Patent Searches***

The use of a standardized set classification codes for describing patents ensures that the most comprehensive preliminary information is available for a patent search

---

<sup>1</sup>In IR terms, patent applications and prior art patents would respectively be referred to as query and its appropriate time-sliced dataset. From these we provide a ranking of prior art patents that have claims which may overlap on the claims of the patent application.

[36]. Moreover, a proper classification search (a search using classification codes) allows a patent researcher to bypass many key problems associated with keyword searching, including [20]:

- *Vague or inconsistent terminology*—frequently, the titles and abstracts of patents are too broad for a keyword search, so coming up with a full set of keywords is challenging.
- *Different meanings in different fields*—keyword terms may describe a wide variety of different concepts, such as the use of “mousetrap” to describe a specific type of logic circuit.
- *Synonyms*—may be simple synonyms, such as using ‘water closet’ to describe a toilet or using ‘a rodent extermination device’ to describe a mousetrap.
- *Foreign Spellings*—A given language could have different spellings to describe a single term, such as sulfur/sulphur, or aluminum/aluminium.
- *Obsolescence*—Given the permanency of the patent collection, certain terms such as “LP”, “hi-fi”, and “water closet” may not longer be the way to describe what a patent covers.
- *Spelling errors and variations*—Patents are not immune to spelling errors or regional variations in terms, adding to the complexity of keyword searches.
- *Acronyms and abbreviations*—some terms use standard—or non-standard—acronyms to describe a given patent, whereas others spell out some or all the terms. The same is true about abbreviations in patent descriptions, claims titles and abstracts.

A patent search using classification codes is particularly critical when looking for design patents, as they pose a unique challenge with limited text, so searches using classification codes provide the most efficient way to search both types of patents. Moreover, a search on utility patents should include design patents as well, since many inventions incorporate both types of patents. As rich in information as classification codes are, little research has been done to examine similarity between codes in a classification hierarchy. Our motivation is to examine this aspect of the classification code structure and see if there are methods, which can be used to make searches on classification codes more effective.

### ***14.1.3 Patent Classification Systems***

Patent-issuing bodies such as the European Patent Office (EPO), the World Intellectual Property Office (WIPO), and the United States Patent and Trade Office (USPTO) manually classify each patent application into one or more of many *classifications* based on the patent’s intended use. The European Patent Office also classifies patents using its own European Classification (ECLA) system, and this is performed by patent experts for European patents, as well as foreign patents. Sub-classifications serve as a more granular categorization of a particular class. The USPTO, for example, classifies each patent into at least one of approximately 470

classes and 163,000 subclasses. Likewise, the WIPO claims over 71,000 separate classifications down to the subgroup level [5]. The ECLA contains over 133,000 entries. Over 100 patent issuing bodies use the IPC, making it the most widely used patent classification system. The EPO and other European national patent issuing bodies also use the ECLA, in addition to the IPC. The USPTO uses the USPC primarily, but US-issued patents reference the IPC system as well.

In an effort to make patent documents accessible for a variety of uses, nearly all patent-issuing bodies have some form of classification of patent utility. This classification allows patent documents to be easily retrieved and identified [36].

For any patent-issuing office to ensure an invention is genuinely novel, it is essential to not only search the patents published by their own office but also those published by patent-issuing offices in other countries. Thus, there is an essential need for a single international classification system to make a search for prior art across the collections of numerous international patent offices more efficient.

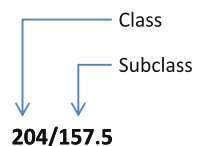
In Sect. 14.3, we investigate the similarity between classification codes in a hierarchy—we examine the relationship between the similarity of codes in the classification code hierarchy and the similarity of the patents containing those codes. This may aid in the discovery of prior art in patents.

#### 14.1.3.1 USPC Classification

The United States Patent Classification (USPC) system is used by the United States Patent and Trademark Office (USPTO), although it was announced in October 2010 that a joint patent classification system between the USPTO and EPO, based on the ECLA was forthcoming [7]. Each of the 163,000 entries consists of a Class and a Subclass. Patent classes and subclasses are organized into a fairly deep hierarchy, although a patent’s classification can be represented with a few levels (top-level class and most-distinct subclass). There may be as many as 14 distinct subclass levels for a given class. Below in Fig. 14.1 is an example illustrating how the USPC system is utilized in a particular patent.

- The highest hierarchical level in the USPC is the Class. For example, Class 204 is associated with patents that demonstrate “Chemistry: Electrical and Wave Energy”, such as for the method of generating and producing ozone.
- Its subclass 157.5 “Oxygen containing product produced” is actually three levels deep in the USPC subclass structure under Class 204. From the definition provided by USPC [31] we see that:
- Subclass 157.5 is a child of subclass 157.4: “Process of preparing desired inorganic material”

**Fig. 14.1** An example of a USPC classification in a patent document



- Subclass 157.4 is a child of subclass 157.15 “Processes of treating materials by wave energy”
- Subclass 157.15 is a child of the Class 204 definition.

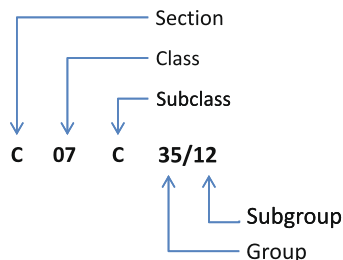
Thus we see that USPC classification codes may appear in rather succinct format in the patent document.

Complicating USPC classifications further is the dynamic nature of how the classification scheme itself evolves. Earlier investigation by Larkey [18] found that a single USPC subclass can have up to 2,000 patents. However, the USPTO attempts to limit the patents in a single subclass to no more than 200 by creating additional subclasses. Periodic reviews of the classification system by the USPTO often result in the restructuring of many subclasses by further dividing, merging, or eliminating subclasses. Additionally, new inventions may require an entirely new set of classes and subclasses to be introduced to accurately describe the invention’s intended use [8].

### 14.1.3.2 IPC Classification

In 1971, twenty-three countries signed the Strasbourg Agreement, a number that has now grown to 61 signatory countries [35]. This Agreement established the International Patent Classification (IPC) under the World Intellectual Property Organization (WIPO), which divides technology into eight discrete Sections. The goal of this Agreement was to overcome the difficulties caused by using diverse national patent classification systems. The resulting classification system was designed to be applicable to both patent and utility model documents. This IPC taxonomy, now in its eighth version, is updated every five years<sup>2</sup> and now consists of over 71,000 codes. Each code is described by a “classification symbol” called an IPC code. A patent is generally assigned to one or more IPC codes that indicate the related technical field or fields the patent covers. These codes are arranged in a hierarchical, tree-like structure with five distinct components [36]. An example of one such IPC code follows in Fig. 14.2 below.

**Fig. 14.2** An example of an IPC classification



<sup>2</sup>The ninth version of the IPC is due to be released on January 1, 2011.

- The highest hierarchical level contains the eight sections of the IPC corresponding to very broad technical fields, labeled A through H. For example, Section C deals with “Chemistry and Metallurgy”.
- Sections are subdivided into classes. The eighth edition of the IPC contains 120 classes. Class C07, for example, deals with “Organic Chemistry”.
- Classes are further subdivided into more than 600 subclasses. Subclass C07C, for example, deals with “Acyclic or Carbocyclic Compounds”.
- Subclasses are then further divided into main groups and subgroups.
- Main group symbols end with “/00”. Ten percent of all IPC groups are main groups. For example, main group C07C 35/00 deals with “Compounds having at least one hydroxy or O-metal group bound to a carbon atom of a ring other than a six-membered aromatic ring”.
- The hierarchy of the subgroups under main groups is designated by dots preceding the titles of the entries. For example, under C07C class 35, we have several hierarchical subclasses, to three subclass levels.
  - C07C 35/02 is the subclass consisting of “Monocyclic compounds”
  - C07C 35/08 is indented to a second level and so is under C07C 35/02 “Monocyclic compounds” and are “Monocyclic compounds containing a six-membered ring”
  - C07C 35/12 is indented to a third level and so is under both C07C 35/02 and C07C 35/08 and includes those “Monocyclic compounds containing a six-membered ring” that consist of “Menthol”.

In our example above, C07C 35/12 represents the IPC classification for the production of Menthol.

- In some versions of the IPC, a series of numbers will follow the subgroup, reflecting the enactment date of the IPC version. ‘20060101’ following the Subgroup indicates a date of January 1, 2006, which is the date that the eighth version of the IPC took effect.

## ***14.1.4 Differences Between Major Patent Classification Systems***

### **14.1.4.1 Difference Between the IPC and USPC**

With more than 163,000 subdivisions, and 460 classes, each subclass of the USPC has a far tighter focus compared with the IPC, which contains less than half as many unique classification codes. Also, the more granular USPC was designed for patent searches whereas the IPC was designed for harmonization between different patent-issuing bodies. Hence, in terms of depth of classification, the USPC usually gives more precise information on the invention’s true purpose. Also the IPC classifies an invention according to its “function” whereas the USPC not only classifies based on the function but also on the industry, anticipated use, intended effect, outcome,



**Table 14.1** IPC/USPC terminology equivalents

IPC	USPC
Section	Discipline
Class	Category of Classes
Subclass	Class
Group	Subclass
Subgroup	Indented Subclass

and structure. Because of this narrow focus, the USPC can be more challenging to search. However, a search across the broader categorization of the IPC code may return a wider variety of patents—including many unrelated ones—in a crowded invention field. Table 14.1 illustrates some of the categorization terminology used with the IPC and its USPC equivalent.

Though the USPTO has developed a concordance between its USPC system and the IPC, the provided concordance tables are not always accurate nor complete since the various classification systems are different and not all revised at the same frequency. Other difficulties with the concordance tables include the different underlying philosophies of the two systems and that a complete one-to-one relationship has not been attained.

In the printed versions, both the IPC and USPC systems hierarchically indent dependent titles with ‘dot’ notation, where the number of dots preceding the subgroup indicates the depth in the subgroup hierarchy. With the USPC system, the first occurring subclass of two subclasses at the same level is always superior. On the other hand, the IPC system does not have a single standard superiority rule as is found in the USPC system. The IPC system utilizes either general placement rules for superiority when no rules are stated in the notes or text of an IPC subclass schedule or its class, or one or more of several particular placement rules that are clearly stated in the notes or text of an IPC subclass’ schedule or its class.

#### 14.1.4.2 Differences Between the IPC and ECLA

The ECLA system, created and used by the EPO, is often used for searching patents issued by European patent-issuing offices. More than 28 million documents can be searched using ECLA symbols, sometimes dating back to 1836 (depending on the country of origin) making it more versatile than the IPC. Additionally, since ECLA contains nearly 133,000 entries—nearly twice as many entries as in the IPC, a higher relative precision in the scope definition for a specific entry can be expected using ECLA symbols instead of with IPC symbols. Additionally, the ECLA is updated monthly, compared with annual updates for the IPC, allowing for classification codes to better match the evolution of technology.

However, not all documents are available which contain the ECLA classification system. On the other hand, the IPC documentation is the largest available classified collection of patent documents—more than 37 million documents are classified with IPC symbols. One limitation with the IPC system is that it does not cover documents

published prior to 1968. One should note that the classification groupings used in the ECLA system is identical to that used in the IPC system, but with increased level of granularity [6].

## 14.2 Related Work Using Classification Hierarchies

Due to the increase in costs of intellectual property litigation and advancements in text mining, a number of techniques have been introduced to examine patent similarity. Initially, much work focused on the categorization of patents into similar groupings. Chakrabarati et al. [3] performed some small-scale tests using Bayesian classification methods and demonstrated that categorization of patents could be improved by using the classifications of cited patents. Larkey [18] was able to improve the precision of patent searches using the  $k$ -Nearest Neighbor approach, but was unable to improve patent categorization. Fall et al. [9] showed how different measures, when indexed against different sections of a patent's corpus can improve results against the IPC system; however, these methods focus more on the examination of appropriate query search terms.

More recently, attention has been focused on evaluating relevance of prior art patents for a given patent application. NTCIR (National Institute of Informatics, Japan) has run several workshops in information retrieval that focused on searches on Japanese patents [10, 23, 24], but the focus was primarily on evaluating patent search terms to determine relevance. One of the tasks of the 2009 and 2010 TREC Chemical Patent tracks sponsored by the NIST (National Institute of Science and Technology) requests participants to provide a ranked list of prior art patents that are most closely related to a given set of patent applications [29]. The effectiveness of the use of IPC classification system in patent retrieval has had mixed results. In TREC-Chem'09, the BiTeM Group discovered that the use of the IPC code actually hindered search quality on chemistry-related patents [12] but did the IPC code did not do so on a broader set of patents from the CLEF-IP 2009 dataset [11]. Earlier research using the IPC at NTCIR found success in its use in comparing patent applications and prior art patents [15, 16]. Also at TREC-Chem'09, the Purdue team found that the IPC aided their ability to retrieve prior art patents for a given patent application [2].

Research in linguistics has focused on evaluating the distance between nodes of hierarchical structures. Shahbaba and Neal [26] have used Bayesian form of multinomial logit (MNL) to improve classification of terms, but this technique requires prior knowledge of correlations between nodes, which is expensive to calculate. WordNet is a lexical database of the English language that has been studied extensively since its introduction in 1985 [1, 22]. It contains groupings of English words into groups of synonyms called synsets. WordNet's purpose is to permit examination of the semantic relations between these synonym groupings. In Sect. 14.3, we develop a WordNet-type structure to examine the semantic relations between classifications. Leacock and Chodorow [19] and Rada [25] have focused on semantic relatedness of WordNet ontologies. In the experiments conducted in this chapter,

we borrow from ontological similarity techniques and extend them to patent classification hierarchies.

Our goal is to examine the usefulness of classification hierarchies within the domain of patent classification code searches. We have designed several methods that we believe will improve search recall and precision, and in the next section, will discuss four hypotheses to examine these methods.

## 14.3 Experimental Design

In this section, we describe how we establish our baseline and develop several hypotheses for examining the role of classification hierarchies in patents. We then discuss our methodologies for conducting our experiments.

### 14.3.1 Data

Our goal is to explore how the hierarchy of IPC codes can be used to produce a ranked list of potentially invalidating patents for a given patent application. We chose to constrain the patents we used in this study to a single domain (chemistry). To accomplish this, we focus on those classes determined by CAS (Chemical Abstract Service) to be related to chemistry [4], but our methodology is also applicable to other domains.

First we created a machine-readable version of the IPC classification, ensuring that each code is represented by its complete hierarchical path. This effort was required as the code's full path was not explicitly available in the patent document. We also separated the first part of the IPC hierarchy, containing the Section, Class, and Subclass, from the second part containing the Group and Subgroup. This separation allows us to apply weights to each of them separately and determine their relative influence on the resulting ranked document list. This information is then inserted into the XML version of the patent document.

The distribution of IPC codes in our dataset is skewed right, with a majority of IPC codes tied to only a few patents, but with some IPC codes having as many as 106 associated patents. The total number of IPC codes in our dataset considered is 26,014, and the mean number of patents listing a given IPC code is 55, while the median number is 18. The IPC codes that are referred to in fewer than 10 and 50 patents take about 38% and 65%, of all IPC codes available in our collection, respectively.

### 14.3.2 Baseline Retrieval and Ranking

We first retrieved a ranked set of up to 1,000 prior art patent documents for each of the 100 patent applications from the PA-Small task of TREC-Chem'09 [29], using

the methods discussed for TREC-Chem Run 1 in [21]. The retrieval methods used to produce these ranked sets did not consider classification codes; instead they rely on text searches against the patent's title, abstract, claims, and description fields. In total we obtain a set of 74,603 distinct retrieved patents for the 100 patent applications. Of these retrieved patents, 18,806, or 25%, are patents that were issued by the EPO and the remaining 55,797 patents were issued by the USPTO. We establish these or ranked sets of retrieved patents as our baseline to explore the four different hypotheses. Of these retrieved prior art patents in the baseline, 2,870 were judged as patents relevant to the corresponding patent application. Our procedure is to re-rank these retrieved patents using our new methods to determine if our methods could be improved solely by examination of the IPC classification system alone. The Indri-based strategy we applied is identical to the one used in [14]. If we are able to significantly improve the ranking of the retrieved patents based on the classification hierarchy alone, we believe that our methods can make a difference to patent retrieval.

### ***14.3.3 Hypotheses***

#### **14.3.3.1 Examination of Classification Hierarchies**

Our first hypothesis is that the use of the information contained in classification hierarchies can aid a patent examiner in finding patents similar to a given patent application. Also we believe that a prior art patent's classification codes need not to be exact matches to those contained in the patent application. However, the closer the classification codes from the patent application and a given prior art patent are in proximity with one another in the classification code hierarchy, the more similar the two patents are. Our reasoning is that the classification hierarchy was implicitly designed with this concept of proximity in mind, and this can be useful in retrieving similar patents even if they use different codes.

#### **14.3.3.2 Examination of the Weighting of Class and Group**

Our second hypothesis is the importance of the second component of the IPC code—Group and Subgroup—is more valuable than the first component—Section, Class and Subclass—of the IPC code in the discovery of prior art patents (see Sect. 14.1.3.2 for a description of the IPC format). Moreover, we wish to determine the ideal ratio between these two components of the IPC to provide the best-ranked patent result set. As mentioned earlier, the first component contains more general information whereas the second component of the IPC code is more specific.

Thus a match, either exact or approximate, on the less-specific first component of the IPC, i.e., the Section, Class, and Subclass, between two patent documents is less likely to be meaningful than a match at the more specific component containing the Group and Subgroup levels. We study this intuition by exploring differing weights on these two different parts of this hierarchy.

### 14.3.3.3 Examination of Primary Classification Codes vs. All Classification Codes

Our third hypothesis is that there is a significant difference between the ranked list of prior art patent documents returned using the primary IPC classification code alone, compared with the use of all IPC classification codes in the prior art patent documents being considered.

### 14.3.3.4 Examination of Primary Classification Codes vs. Our Baseline

Next, we examine the position of the IPC code within a patent. WIPO, in their 38th Session, released an “Order of Recording of Classification Symbols”. In this Order, they state “Classification symbols representing invention information, of which that symbol which most adequately represents the invention should be listed first” [33]. A follow-up report released in their 39th Session in 2007 indicated that 32 of 34 patent issuing bodies surveyed were following this approach [34]. But, to our knowledge, no study has examined the effectiveness of this approach using the IPC. Thus, our fourth hypothesis is that the use of the primary, or first-listed, classification code for the two patents will produce a significant difference in the ranked list of patent documents relative to those returned in our baseline.

## 14.3.4 Methodology

For the experiments we conducted on classification hierarchies in this chapter, we focused our examination on the IPC. Our choice of the IPC is two-fold; first, the IPC is the most widely used, being used by over 100 patent-issuing bodies to classify their patents. Second, with the IPC, the Group and Subgroup are far more specific (whereas the Category, Class and Subclass are more general); this allows us to see how each affect the nature of searches based on the classification code and allows us to determine the effects of classification hierarchies.

We conducted the experiments presented below using the XML-formatted TREC-Chem’09 dataset [29]. After explicitly deriving and inserting a new XML field representing the hierarchical path to each classification code assigned to the document, we used Indri [27] to index, retrieve, and rank our documents. Although patent documents contain a number of fields applicable for searches, our focus here is to examine the classification hierarchy’s utility in this process, so we focus on different aspects the use of classification codes in searches. For example, the IPC code C07C 35/12, shown in Sect. 14.3.2, would be represented in XML as

```
<ipc-path>  
<class-path> C07C </class-path>  
<group-path> 35:00:35:02:35:08:35:12 </group-path>  
</ipc-path>
```

These XML tags are then added for each IPC code appearing in our patent collection. Documents are then compared for similarity based on their IPC code fields; we use Indri for this—the reader is directed to [14] for an explanation of Indri’s role in a similar experiment. Preliminary examination of Indri’s techniques of evaluating hierarchical structures such as classification codes demonstrate that two codes that are more closely related (such as a child, parent, or sibling) patent in the hierarchy, rank more highly than those which are more distantly related.

## 14.4 Results

### 14.4.1 Measures

**Mean Average Precision (MAP)** In our experiments, we obtain a ranked sequence of patents from Indri, and therefore it is desirable to consider the order in which the returned patents are presented. Average precision emphasizes ranking relevant documents higher. It is the average of precisions computed at the point of each of the relevant documents in a ranked sequence of results matching our patent application. See [30] for additional information on MAP.

**Recall Score for the Top 100 Ranked Patents in Our Retrieval List (Recall@100)** Our interest in this metric is based on our understanding that it is rarely practical to examine more than 100 patents for a given patent application. Moreover, MAP and nDCG consider both Recall and Precision, but arguably recall is the more important metric in patent examinations.

**Normalized Discounted Cumulative Gain (nDCG)** This measure uses a graded relevance scale of documents and measures the usefulness (*gain*) of a given document as determined by its position in the entire list of retrieved results. Like MAP, this gain is cumulative—however, as we move down our list of results, the gain obtained by finding a relevant document is discounted further. See [17] for additional information on nDCG and [28] for additional discussion on how nDCG and MAP apply in different retrieval situations.

### 14.4.2 Examination of Classification Hierarchies

For our first hypothesis, we examined if the application of the classification hierarchy provides a more meaningful set of results than if the classification hierarchy is not considered. The results based on our 100 patent application queries show that the classification hierarchy does have an impact. The resulting MAP, Recall@100, and nDCG scores increased by 29%, 37% and 27% respectively. The large increase in recall for the first 100 ranked prior art patents demonstrates how the re-ranking

**Table 14.2** Results from our use of the classification hierarchy to re-rank patents

Metric	Baseline	Using Classification Hierarchy
Num Patents Returned	93676	93676
Num Rel Patents Ret	1118	1121
MAP	0.0485	0.0626*
Recall@100	0.1888	0.2585*
nDCG	0.2245	0.2844*

method does have a significant effect. Table 14.2 shows the results from this test. An asterisk (\*) next to the number indicates a significant improvement over our baseline at a  $p < 0.05$  level of significance.

We discovered a slight increase in the number of patents judged as relevant (from 1,118 to 1,121), even though we are only employing a re-ranking function on a closed domain of patents. Three patents included in the original ranking for other patent applications were examined by our function and determined to be relevant to an additional patent application. This is likely because the three additional patents, after our re-ranking approach was applied, had surpassed a cutoff threshold and were then included for a search on one of the other patent applications. This demonstrates—albeit in a small way—that the use of distances in the classification hierarchy can potentially broaden our ability to retrieve relevant patents in a prior art search.

### 14.4.3 Examination of the Weighting of Class and Group

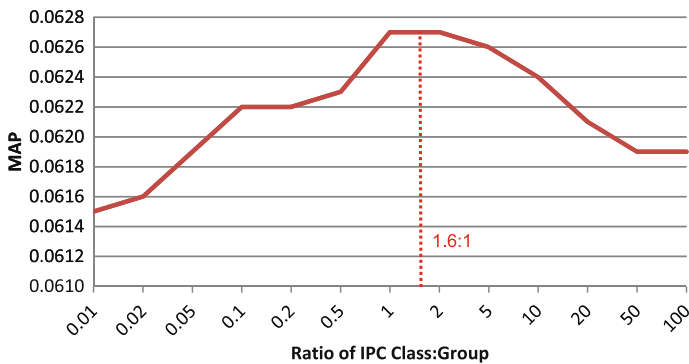
For our second hypothesis, we seek to determine which part of the IPC system—the first, more general part (denoted as the “Class” portion of the IPC) or the second more specific part that includes the group and subgroup (denoted as the “Group” portion of the IPC), affect our metrics more. To accomplish this, we applied a range of different weight ratios between the two parts, ranging from 1:100 to 100:1, and examined the resultant metrics. A subset of the weight ratios and their results appear in Table 14.3.

**Table 14.3** Results of different weights applied to parts of the IPC classification

Metric	Class:Group 1:1	Class:Group 1:2	Class:Group 1:100	Class:Group 2:1	Class:Group 100:1
Rel. Patents Ret	1121	1121	1119	1121	1120
MAP	0.0626	0.0623	0.0615	0.0627	0.0619
Recall@100	0.2585	0.2564	0.2487	0.2592	0.2511
nDCG	0.2844	0.2823	0.2791	0.2849	0.2831

**Table 14.4** Comparison of the optimal weighting between the two parts of the IPC classification and our baseline

Metric	Baseline	Using Class:Group Ratio of 1.6:1	Improvement over Baseline
Num Patents Returned	93676	93676	–
Num Rel Patents Ret	1118	1121	0.3%
MAP	0.0485	0.0629*	29.7%
Recall@100	0.1888	0.2595*	37.4%
nDCG	0.2245	0.2852*	27.0%



**Fig. 14.3** The effect of the Class:Group ratio on MAP

Through further experimentation with the weighting of these two parts of the IPC classification code, we did not discover any significant difference using the weighted methods; in other words, using weights that provided the best result did not significantly differ from the non-weighted ratio (reported in Table 14.2) at the  $p < 0.05$  level of significance. The improvement over our baseline was significant, however: the best score is generated from a ratio between Class:Group of 1.6:1, giving us a 27% improvement, for example, in nDCG. In Table 14.4, we compare the metrics for the best result with that of our baseline result. An asterisk (\*) next to the number indicates a significant improvement over the baseline at a  $p < 0.05$ .

Figures 14.3, 14.4 and 14.5 illustrate the effect of the ratio on our MAP, Recall@100, and nDCG results respectively. In Fig. 14.3, MAP peaks where the Class:Group ratio is 1.6:1, thus indicating that when the weight ratio between the first section of the IPC (containing Section, Class and Subclass) is 1.6 times the weight of the second section (containing Group and Subgroup), MAP is maximized. Figure 14.4 shows that recall@100 also peaks near the point where both parts of the IPC code are considered equally as well, although this peak is not pronounced relative to the other weighting combinations we have considered.



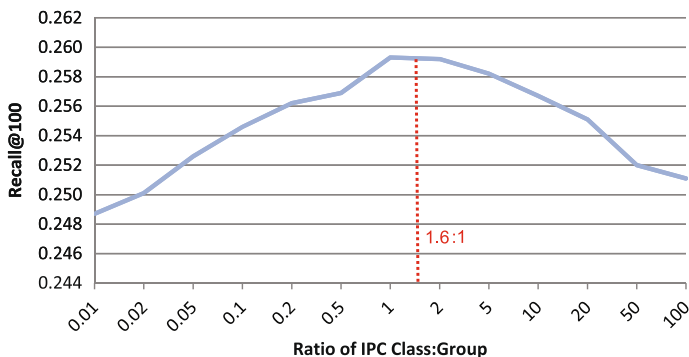


Fig. 14.4 The effect of the Class:Group ratio on Recall@100

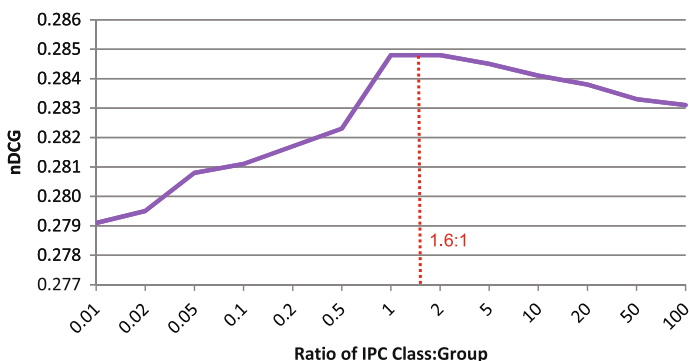


Fig. 14.5 The effect of the Class:Group ratio on nDCG

### 14.4.4 Examination of Primary Classification Codes

In our third hypothesis, we examined whether the use of this primary IPC code for each of our patent documents would show a similar improvement as we found in our first hypothesis, where we used all IPC codes. In our fourth hypothesis, we examined whether the primary IPC code alone would provide a significant improvement over our baseline. We found that the use of the primary IPC code significantly improved performance over the baseline, but the use of the primary IPC code alone lags behind the performance obtained using all IPC codes in patent documents. Table 14.5 illustrates the results we obtained for the use of the primary classification code and its performance against the baseline. An asterisk (\*) next to the number indicates a significant improvement over the baseline at a  $p < 0.05$  level of significance; a plus sign (+) indicates a significant improvement over the use of the primary IPC code alone at a  $p < 0.05$  level of significance.

From Table 14.5, we observe that the primary classification code provides an increase over the baseline results, but this increase is not as large as when all classification codes are used. With  $n$  classification codes in the patent application and  $m$

**Table 14.5** Results from considering the primary IPC classification alone versus considering the full set of IPC classifications to re-rank patents

Metric	Baseline	Using Primary IPC Classification Code Only	Using All IPC Classification Codes
Num Patents Returned	93676	93676	93676
Num Rel Patents Ret	1118	1118	1121
MAP	0.0485	0.0589*	0.0626*+
Recall@100	0.1888	0.2137*	0.2585*+
nDCG	0.2245	0.2516*	0.2844*+

classification codes in a prior art patent, there are  $n \times m$  possible similarity scores to consider for each target-prior art patent combination. When evaluating primary classification codes only, there is only one possible combination of codes to consider for each target-prior art patent combination. Since we are ranking only the maximum similarity score for each target-prior art patent combination, having a larger number of classification code pairings to consider and rank ensures that using all codes will do as well or better than using the primary classification code alone. This result also concurs with the results from a far more limited USPC dataset we had discovered in previous research [13].

In summary, we began with baseline sets of up to 1,000 patent documents for each of our 100 patent applications and then re-ranked these sets using the methods involving the IPC code hierarchy as explained above. Results indicate that patents with more closely related classification codes also are more similar. One limitation of our work is that the initial retrieval set for the 100 queries were already constrained to those results retrieved by our baseline strategy. It is possible that several relevant documents were missed by these methods. Our re-ranking efforts are not designed to improve recall and thus inherit the same limitations. Another likely improvement is to examine the ECLA classification system instead of the IPC or, better yet, to use a weighted balance of IPC, USPC and ECLA codes instead of relying on IPC codes alone, thus taking advantage of the strengths of each classification system.

## 14.5 Conclusion

We have described the purpose of classification hierarchies and explained three major classification systems used in evaluating prior art in patents. We then examined the role of classification code hierarchies in a patent dataset and explored four hypotheses. The test of our first hypothesis demonstrated that the classification code hierarchy can be utilized to significantly improve the ranking of retrieved prior art patents. The test of our second hypothesis showed that both parts of the IPC classification code system are important to patent retrieval, although the categorical portion

containing the Section, Class and Subclass are slightly more important to exploiting a classification hierarchy than the hierarchical portion containing the Group and Subgroup. The test of our third and fourth hypotheses demonstrated that the use of all classification codes for a given patent application and all matching prior art patents provides significantly better results than using the primary classification code alone; however, the use of the primary classification code by itself can still significantly improve results over our baseline.

The designs of the experiments in this chapter have demonstrated that classification code hierarchies can be used as a boosting measure to re-rank patents selected using keyword searches or based on other criteria. By ranking the most relevant patents at the top of the list, a patent examiner can see those patents more likely to infringe on a given patent application.

## References

1. Budanitsky A, Hirst G (2001) Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In: Workshop on WordNet and other lexical resources, second meeting of the NAACL, Pittsburgh, USA
2. Cetintas S, Luo S (2009) Strategies for effective chemical information retrieval. TREC 2009 notebook
3. Chakrabarti S, Dom B, Indyk P (1998) Enhanced hypertext categorization using hyperlinks. In: Tiwary A, Franklin M (eds) Proceedings of the 1998 ACM SIGMOD international conference on management of data, SIGMOD '98, Seattle, Washington, United States, June 01–04, 1998. ACM, New York, pp 307–318
4. Chemical Abstract Service (CAS) IPC8 guaranteed coverage. <http://www.cas.org/expertise/cascontent/caplus/patcoverage/ipcguar8.html>. Accessed 11 Dec 2010
5. European Patent Office (EPO) Coverage of classifications (IPC and ECLA). [http://ep.espacenet.com/help?topic=classesqh&locale=en\\_EP&method=handleHelpTopic](http://ep.espacenet.com/help?topic=classesqh&locale=en_EP&method=handleHelpTopic). Accessed 11 Dec 2010
6. European Patent Office (EPO) IPC (International Patent Classification). <http://www.epo.org/patents/patent-information/ipc-reform.html>. Accessed 11 Dec 2010
7. European Patent Office (EPO) The USPTO and EPO work toward joint patent classification system. <http://www.epo.org/topics/news/2010/20101025.html>. Accessed 11 Dec 2010
8. Falasco L (2002) United States patent classification: system organization. *World Pat Inf* 24(1):111–117
9. Fall CJ, Toresvari A, Benzineb K, Karetka G (2003) Automated categorization in the international patent classification. *SIGIR Forum* 37(1):10–25
10. Fujii A, Iwayama M, Kando N (2007) Overview of the patent retrieval task at the NTCIR-6 workshop. In: Proceedings of NTCIR-6 workshop meeting, pp 359–365
11. Gobeill J, Teodoro D, Patschek E, Ruchi P (2009) Exploring a wide range of simple pre and post processing strategies for patent searching in CLEF IP 2009. CLEF 2009 Working Notes
12. Gobeill J, Teodoro D, Patschek E, Ruchi P (2009) Report on the TREC 2009 experiments: chemical IR track. TREC 2009 notebook
13. Harris CG, Foster S, Arens R, Srinivasan P (2009) On the role of classification in patent invalidity searches. In: Proceeding of the 2nd international workshop on patent information retrieval, PaIR '09, Hong Kong, China, November 06, 2009. ACM, New York, pp 29–32
14. Harris CG, Arens R, Srinivasan P (2010) Comparison of IPC and USPC classification systems in patent prior art searches. In: Proceeding of the 3rd international workshop on patent information retrieval, PaIR '10, Toronto, Canada, October 26, 2010. ACM, New York

15. Itoh H (2005) Patent retrieval experiments at ricoh. In: Proceedings of NTCIR workshop 5 meeting, December 6–9, 2005, Tokyo, Japan
16. Konishi K (2005) Query terms extraction from patent document for invalidity search. In: Proceedings of NTCIR workshop 5 meeting, December 6–9, 2005, Tokyo, Japan
17. Jarvelin K, Kekalainen J (2002) Cumulated gain-based evaluation of IR techniques. *ACM Trans Inf Syst* 20(4):422–446
18. Larkey LS (1999) A patent search and classification system. In: Proceedings of the fourth ACM conference on digital libraries, DL '99, Berkeley, California, United States, August 11–14, 1999. ACM, New York, pp 179–187
19. Leacock C, Chodorow M (1998) Combining local context and WordNet similarity for word sense identification. In: Fellbaum C (ed) *WordNet: an electronic lexical database*. MIT Press, Cambridge, pp 265–283
20. Mackey T (2005) Patent searching using the united states patent classification system. Presentation at the 28th Annual PTDL Training Seminar, April 2005
21. Mejova Y, Ha Thuc V, Foster S, Harris CG, Arens R, Srinivasan P (2009) TREC blog and TREC chem: a view from the corn fields. *TREC 2009 notebook*
22. Miller GA (ed) (1990) *WordNet: an on-line lexical database*. *Int J Lexicogr* 3(4):235–312
23. Nanba H, Fujii A, Iwayama M, Hashimoto T (2008) Overview of the patent mining task at the NTCIR-7 workshop. Proceedings of the seventh NTCIR workshop meeting
24. Nanba H, Fujii A, Iwayama M, Hashimoto T (2010) Overview of the patent mining task at the NTCIR-8 workshop. Proceedings of the eighth NTCIR workshop meeting
25. Rada R, Mili R, Bicknell E, Blettner M (1989) Development and application of a metric on semantic nets. *IEEE Trans Syst Man Cybern* 19(1):17–30
26. Shahbaba B, Neal R (2007) Improving classification when a class hierarchy is available using a hierarchy-based prior. *Bayesian Anal* 2(1):221–238
27. Si L, Jin R, Callan J (2002) A language modeling framework for resource selection and results merging. In: Proceedings of the eleventh international conference on information and knowledge management, McLean, VA, Nov 4–9, pp 391–397
28. Stanford Natural Language Processing Group. Evaluation of ranked retrieval results. <http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-ranked-retrieval-results-1.html>. Accessed 11 Dec 2010
29. Text Retrieval Conference (TREC). <http://trec.nist.gov/>. Accessed 11 Dec 2010
30. Turpin A, Scholer F (2006) User performance versus precision measures for simple search tasks. In: Proceedings of the 29th annual international SIGIR conference on research and development in information retrieval, Seattle, USA, Aug 06–11, 2006, pp 11–18
31. United States Patent Office (USPTO) Manual of patent classification (Class 204). <http://www.uspto.gov/web/patents/classification/uspc204/sched204.htm>. Accessed 11 Dec 2010
32. United States Patent Office (USPTO) Manual of patent examining procedure (MPEP), July 2008
33. World Intellectual Property Office (WIPO) (2006) Order of reporting of classification symbols. In: Report of the thirty-eighth session of the IPC, IPC/CE/38/6
34. World Intellectual Property Office (WIPO) (2007) Order of reporting of classification symbols. In: Report of the thirty-ninth session of the IPC, IPC/CE/39/4
35. World Intellectual Property Office (WIPO) Contracting Parties of the Strasbourg agreement. [http://www.wipo.int/treaties/en/ShowResults.jsp?lang=en&treaty\\_id=11](http://www.wipo.int/treaties/en/ShowResults.jsp?lang=en&treaty_id=11). Accessed 11 Dec 2010
36. World Intellectual Property Office (WIPO) Frequently asked questions about the international patent classification (IPC). <http://www.wipo.int/classifications/ipc/en/faq/index.html>. Accessed 11 Dec 2010

# Part V

## Semantic Search

The term ‘semantic search’ is used to describe search that goes beyond simple matching of query and document keywords. In practice, as noted previously in Chap. 2, the term semantic search in patent retrieval is applied to two different forms of enhanced search. First are techniques that rely on compression or clustering of the underlying term/document matrix, like Latent Semantic Analysis and related techniques. These might be called techniques that rely on emergent properties of the data. Second are techniques which use ontologies, thesauri or other taxonomic classifications to augment the process of identifying and indexing potential search terms extracted from the documents.

In a sense, all searching of patents is semantic, since all patents are categorized by the taxonomic IPC codes if nothing else, and these codes are often used as part of the search by patent searchers. Furthermore, in practice, there is no solid line between the two forms of semantic search, since many in the second, ontologically based group, acquire the ontology data at least in part via machine learning techniques related to the first (emergent) techniques.

This section will discuss the use of semantic search in patent retrieval. Because there is little work to report in practical experience with patent data (although progress is being made, for example the Fraunhofer SCAI group at TREC-CHEM—see Chap. 5 in this volume), the related area of scientific paper search, where semantic search is much more developed, will be reported as well. Of course, scientific paper search is an integral part of many forms of patent search.

We have entitled this part ‘Semantic Search’, and while that is a useful and reasonable title for the part, the five chapters represent only one of the two meanings for semantic search in common use in the patent search community. The chapters in this part are all about combining some form of generally statistical analysis of incoming documents, with other external, generally hand crafted, forms of knowledge or data (e.g. ontologies, thesauri or classification schemes like the IPC). We had hoped to include at least one chapter on the use of purely emergent semantics, for example, Latent Semantic Analysis (LSA). However this proved impossible.

The key challenge for patent information retrieval is to move semantic search from the feasibility study stage of technology to the stage of proven improvement of effectiveness of search. In other words, moving from showing whether or not in principle it is possible to use external knowledge sources (of whatever kind) in search systems to actually delivering better results to professional patent searchers.

One of the issues at the time of writing (2011) is that the use of automated and semi-automated ontology acquisition and maintenance tools, often making use of statistical machine learning, for example, has made the distinction between the two forms of semantics far less clear cut than it was a few years ago.

The chapter by Cunningham and colleagues, which begins this part, focusses on the use of deeper analysis of patent documents to facilitate more effective indexing for search. Briscoe and colleagues describe a new approach to analysing and searching scientific papers, which are the key non-patent resource for all forms of search that need to go beyond patent data. The domain of chemical structure information, an especially important form of knowledge in the patent world, is covered by Holliday and Willett with an overview of the challenges presented by this field and a summary of the approaches to these challenges. Pesenhofer and colleagues describe new methods to use the free-to-access Wikipedia Science Ontology and Linked Open Data to add additional terms to patents in order to address what is sometimes called the vocabulary problem: that searchers often miss relevant documents because the document does not use the same words as the searcher to describe the topic. Finally Nanba and colleagues also focus on the vocabulary problem, and review a new approach to it based on the differences between patent and scientific documents.

# Chapter 15

## Information Extraction and Semantic Annotation for Multi-Paradigm Information Management

Hamish Cunningham, Valentin Tablan, Ian Roberts, Mark A. Greenwood, and Niraj Aswani

---

Why ‘multiparadigm’? Why ‘information management’, instead of the more familiar ‘information retrieval’ or ‘search’? Is our terminology, as suggested by one of our reviewers, ‘PR driven’?! At one level, perhaps, our title does indeed reflect the increasing penetration of short-termist market-oriented motivations into science and engineering (itself a part of the wider subjection of all areas of public life to corporate profit—see for example [3]). The work reported here was funded in part by a company with a close eye on its commercial potential, and we were concerned to describe that potential in our publications. There are also, however, two substantive points which we wanted to make, which are worth explaining in a little more detail. First, we believe that Mimir is distinctive in combining three types of indexing under one roof—hence *multiparadigm*—full text/boolean; conceptual/semantic; annotation (graph) structure. It is not the case that this combination is either commonplace or straightforward, and we are hopeful that the work will be influential as a result. Second, the technology suite into which Mimir fits is not just about indexing (or information retrieval as commonly defined)—hence *information management*—including as it does both GATE Teamware, a workflow-based distributed collaborative manual annotation and annotation management platform, and OWLIM, a semantic repository which is increasingly used for curated data sets as well as indices (examples in the UK include the BBC, the Press Association and the National Archive). We feel, therefore, that Mimir is an appropriate name for this enterprise.

---

H. Cunningham (✉) · V. Tablan · I. Roberts · M.A. Greenwood · N. Aswani  
Department of Computer Science, University of Sheffield, Sheffield, UK  
e-mail: [H.Cunningham@dcs.shef.ac.uk](mailto:H.Cunningham@dcs.shef.ac.uk)

V. Tablan  
e-mail: [V.Tablan@dcs.shef.ac.uk](mailto:V.Tablan@dcs.shef.ac.uk)

I. Roberts  
e-mail: [I.Roberts@dcs.shef.ac.uk](mailto:I.Roberts@dcs.shef.ac.uk)

M.A. Greenwood  
e-mail: [M.Greenwood@dcs.shef.ac.uk](mailto:M.Greenwood@dcs.shef.ac.uk)

N. Aswani  
e-mail: [N.Aswani@dcs.shef.ac.uk](mailto:N.Aswani@dcs.shef.ac.uk)

**Abstract** This chapter describes the development of GATE Mímir, a new tool for indexing documents according to multiple paradigms: full text, conceptual model, and annotation structures. We also present a usage example for patent searchers covering measurements and high-level structural information which was automatically extracted from a large patent corpus.

## 15.1 Introduction

Dear Reader,

The gentle tinkling noise that you can hear in the background is the sound of genre expectations shattering. This is not an ‘*intellectual property*’ paper (indeed I<sup>1</sup> am uncertain that such a thing really exists—how could *intellect* be *property*?). As Glyn Moody<sup>2</sup> points out, Turing’s results imply the atomicity of the digital revolution, and consequently it seems likely that the electronic genie is now so far out of the ‘rights’ bottle that all bit stream representations of our human achievements will follow into the realm of openness and cooperative enterprise sooner or later. Nor is this an *information retrieval* paper, at least not in the well-behaved sense of positing a hypothesis about model performance within a particular set of parameters and then testing and drawing some familiar variant of an ROC curve to show how well our hypothesis applies in the context of some particular data set.

We will break the expectations of patent searchers by paying little attention to the particular needs of that community (although the work we report was initially applied to patents and is likely to have benefits there), and perform similar violence to the expectations of IR researchers by making a fairly rudimentary evaluation. Now that only the curious are still reading, I can appeal to you as a kindred spirit.<sup>3</sup> Join me in a short history of some technological developments that my colleagues and I have had the pleasure of making over the last few years. I promise not to tell anyone about your paradigm-shifting deviance if you’ll extend the same courtesy to me.

The paper covers work on two areas. First, the integration of standard information retrieval techniques with semantic annotation and information extraction work in order to deliver search capabilities that may be more flexible and interactive than previously. Second, on scalability via distributed processing and efficient indexing. It is structured as follows:

- we begin in Sect. 15.2 with some context and terminology relating to both the characteristics of patent searching and the text mining technology from which Mímir has developed
- in Sect. 15.3 we present the design and implementation of Mímir

---

<sup>1</sup>In the interests of protecting the innocent the first author lays claim to the introduction.

<sup>2</sup><http://opendotdotdot.blogspot.com/>.

<sup>3</sup>I used to hope that as time passed I would become older and wiser, but it seems that in fact I just become odder and wider.



- Sections 15.4 and 15.5 go on to describe the semantic annotation of patent documents with general categories such as bibliographic references or sectioning and specific data on measurements that appear in the texts
- Section 15.6 gives an extended example of the type of multi-paradigm search process that is possible as a result of pushing the annotations described in 15.4 and 15.5 into a Mimir index server
- Section 15.7 wraps up with discussion of the main achievements described within this chapter

## 15.2 Background

### 15.2.1 *Semantic Annotation*

Semantic annotation is the process of attaching metadata tags and/or ontology classes to text segments, as an enabler for knowledge access and retrieval tools. Automatic annotation is carried out by employing Information Extraction (IE) [4] techniques, which recognise automatically instances of a given set of events, entities or relationships. From an algorithmic perspective, IE approaches fall in to two broad categories: manually engineered ones (frequently based on pattern-matching like rules, see e.g. [13]) and machine learning ones (see e.g. [2, 12]). Rule-based approaches are more suitable where a carefully engineered, high precision system is needed and there are not sufficient training data for a machine learning approach to be successful. From an operational perspective, IE tools can be deployed in both fully and semi-automatic applications (where users can inspect and, if needed, correct the automatically created metadata). In general, fully automatic methods are preferred when the volume of data is too large to make human post-annotation practicable, as is the case with patents.

### 15.2.2 *Patents*

Patents are an important vehicle for the expression of corporate strife, and this importance is increasing in the current intensification of international competition. When researching new product ideas or filing new patents, inventors and patent attorneys need to retrieve all relevant pre-existing know-how and/or exploit and enforce patents in their technological domain. This process may be hindered, however, by a lack of rich metadata, which if present, would allow powerful concept-based searches to complement the traditional keyword-based approaches.

Patent searchers require high recall methods, capable of operating robustly on large volumes of data. Much early IE research was carried out on smaller datasets from narrower domains, often news articles [2, 9, 14]. A challenge addressed more recently is in scaling up these methods to deal with the diversity and volume of patent data.

Applications of IE to patent annotations are quite scarce, mostly focusing on optical character recognition (OCR) and text classification, whilst only briefly discussing the importance and challenges of identifying references to figures and claims in patents. In this area, [11] carried out a small feasibility study using the Xerox language processing tools. The PatExpert project [15] has developed some content extraction components based upon deeper linguistic analysis than the approach proposed here.

### 15.2.3 ANNIE and ANNIC

In 2007 we began work on adapting an IE and semantic annotation system to patent data. This system (ANNIE, A Nearly-New IE pipeline) is part of GATE (<http://gate.ac.uk/> [6, 7]), which also includes a diverse set of development tools for language processing R&D. One such tool is ANNIC (ANNotations In Context), which predates the work described here [1]. ANNIC was designed to support the development of finite state transduction patterns in GATE's JAPE language.<sup>4</sup> ANNIC is used to search corpora that have been both annotated with GATE and indexed using Lucene.<sup>5</sup> Users make searches based in a query language very similar to JAPE and are presented with a results summary similar in form to KWIC (Key-Words In Context) tools: the portions of text that match the query form a column down the centre of the screen and are preceded and followed by the proximate text on either side.

ANNIC was designed as a development tool, not as an end-user tool, and is tightly integrated within GATE Developer (a specialist tool for R&D workers), and is inefficient beyond the range of a few hundred documents. We had no intention of proposing the tool as appropriate for patent searchers, but by chance we used it to demonstrate some of the IE work to a patent search expert group. The feedback from this group was very positive, and we were commissioned to produce a version of ANNIC that would scale to a one terabyte plain text database of patent documents—hence Mímir, to whose design we now turn.

## 15.3 GATE Mímir—A Multiparadigm Index

Mímir<sup>6</sup> is a multi-paradigm information management index and repository which can be used to index and search over text, annotations, semantic schemas (ontologies), and semantic metadata (instance data). It allows queries that arbitrarily mix full-text, structural, linguistic and semantic queries, and that can scale to gigabytes of text.

---

<sup>4</sup>JAPE is a regular expression based language for matching annotations—see <http://gate.ac.uk/userguide/chap:jape>.

<sup>5</sup><http://lucene.apache.org/java/>.

<sup>6</sup>Old Norse “*The rememberer; the wise one*”.

### 15.3.1 What Is in a Mimir Index?

A typical semantic annotation project deals with large quantities of data of differing kinds. Mimir provides a framework for implementing indexing and search functionality across all these data types. The data types currently supported within a Mimir index are listed below in the order of increasing information density.

**Text** All documents have a textual content.<sup>7</sup> Support for full text search represents the most basic indexing functionality and it is required in most (if not all) cases. Even when semantic annotation is used to abstract away from the actual textual data, the original content still needs to be accessible so that it can be used to provide textual query fragments in the case of more complex conceptual queries.

Mimir uses inverted indexes<sup>8</sup> for indexing the document content (including additional linguistic information, such as part-of-speech or morphological roots), and for associating instances of annotations with the positions in the input text where they occur. The inverted index implementation used by Mimir is based on MG4J.<sup>9</sup>

**Annotations** The first step in abstracting away from plain text document content is the production of *annotations*. Annotations are metadata associated with text snippets in the documents. Typically an annotation is described by:

- the document it belongs to;
- the start and end offsets of the referred text snippet;
- the annotation type (a textual label or an URI);
- an arbitrary set of <feature, value> pairs.

An annotation index supports a more generic search paradigm. Depending upon the type of annotations available, the user can search across different dimensions. For example, if we suppose that all words in the indexed documents are annotated according to their part of speech, then one could search for sequences of type {Determiner}{Adjective}{Noun}, which would match phrases like *The red car* or *The new method*, etc. When the annotations are semantically richer, this new search paradigm gains more representational power. If, for example, the documents are annotated with occurrences of Person, Location, Organization entities, then searches like {Person}, CEO of {Organization}, based in {Location} become possible.

---

<sup>7</sup>Although the focus is currently on indexing text documents, specifically patents, it would be perfectly feasible to associate annotations and KB data with multimedia documents, where offsets may refer to time spans in videos or areas of an image etc.

<sup>8</sup>*Inverted Indexes* are data structures traditionally used in Information Retrieval to support indexing of text.

<sup>9</sup><http://mg4j.dsi.unimi.it/>.

**Knowledge Base Data** Knowledge Base (KB) data consist of an ontology populated with instances. The ontology represents the data schema and comprises a hierarchy of class types, and a hierarchy of properties that are applicable between instances of classes. The instance data represent facts that are known to the system and are typically, or at least partially, derived from the semantic annotation of documents. KB data are used to reach a higher level of abstraction over the information in the documents and enables conceptual queries such as measurement ranges. A KB is required for answering such queries as they may often involve converting from one measurement unit into another, and reasoning about scalar values.

A KB that is pre-populated with appropriate world knowledge can perform other generalisations that are natural to humans users, such as being able to identify Vienna as a valid answer to queries relating to Austria, Europe or the Northern Hemisphere.

Mímir uses a KB to store some of the information relating to annotations. The links between annotations, the textual data, and the KB information are created by the inclusion into the text indexes of a set of specially-created URIs that are associated with the annotation data. Furthermore, URIs of entities from the KB can be stored as annotation features.

KBs are typically represented as a collection of triples that are kept in highly-specialised and optimised triple stores; using standards such as RDF or one of the versions of OWL.<sup>10</sup> The implementation used by Mímir is based on ORDI and OWLIM.<sup>11</sup>

### 15.3.2 Searching Mímir Indexes

From a user's point of view, Mímir is a tool for searching a collection of semantically annotated documents. It provides facilities for searching over different views of the document text, for example one can search the document's words, the part-of-speech of those words, or their morphological roots. As well as searching the document text, Mímir also supports searches over the documents' semantic annotations; where queries are based on annotation types and restrictions over the values of annotation features. These different search paradigms can be combined freely into complex queries, with support for sequences, repetitions, and Boolean operators.

A search session entails the formulation of a query, running the query with the Mímir query engine, and then consuming the results.

There are two different methods for constructing Mímir queries:

**Query Language:** A simple language has been defined that allows the formulation of Mímir queries using plain text.

<sup>10</sup>See <http://www.w3.org/RDF/> and <http://www.w3.org/TR/owl-features/>.

<sup>11</sup>See <http://www.ontotext.com/ordi/> and <http://www.ontotext.com/owlim/>.

**Java API:** The Mimir Java API defines a set of classes that represent query nodes. Each class corresponds to a type of query that Mimir supports. Individual nodes, representing sub-queries, are combined to form a query tree which embodies a more complex query. The node for the root of the query tree can then be used to execute the query through the Mimir query engine. This format is always used internally by Mimir to represent queries; queries sent in textual form (using the query language) are first converted to a tree of query nodes, and then executed.

There are three different methods for searching with Mimir:

**Web Interface:** When run as a web application, Mimir exposes a GWT- (Google Web Toolkit) based web interface that can be used from any browser. This is the simplest (and most user-friendly) way to access the search functionality of Mimir.

**Java API:** When Mimir is embedded into another Java application the Mimir search API can be used to construct queries, execute them, and process the results.

**Web Service:** When Mimir is run as a web application, a RESTful web service is published that allows the formulation of queries (using the query language), the execution of queries, and the retrieval of results.

Whilst this plethora of query building and search facilities makes Mimir extremely flexible it is unlikely that most patent searchers will need to venture further than entering queries into the web interface (or some other user interface built on top of one of the other search APIs). Given this reasoning, the rest of this section will focus on constructing queries using the plain text query language. For the adventurous, full details of the Java API and Web Service interface can be found in [10].

### 15.3.2.1 Constructing a Query

Mimir queries consist of one or more sub-queries linked by operators. The rest of this sections details the different query types and the operators that can be used to combine them to form more complex queries.

**String Queries:** The simplest form of query is a query term. This will match all occurrences of the query term in the indexed documents.

If the Mimir index being interrogated includes multiple string indexes, then the particular index to be searched can be specified by prefixing the query term with the index name and a colon, for example the query *'root:be'*<sup>12</sup> will match all morphological forms of the verb *to be*. If the name of the string index is omitted, then the first configured index is used. By convention (reflected in the default Mimir configuration) the first string index is used to store the terms text, so the default behaviour is to search over the document text, as expected.

---

<sup>12</sup>This assumes that an index named `root` exists, and was used to store the morphological root of the words.

**Table 15.1** Escaping reserved constructs in the Mimir query language

Reserved input	Escaped form
{, }	\{, \}
(, )	\(, \)
[, ]	\[, \]
:	\:
+	\+
	\
&	\&
?	\?
\	\\
.	\.
"	\"
=	\=
IN	"IN"
OVER	"OVER"
OR	"OR"
AND	"AND"

Some words are part of the query language definition so they cannot be used directly as query terms. If that is desired, then these constructs must be escaped as shown in Table 15.1.

Annotations Queries: If annotations were indexed then Mimir allows searching for annotation-based patterns. An annotation is a piece of metadata associated with a text segment. When indexed in Mimir, annotations are defined by:

- *type*: a string value
- *start and end offsets*: two numeric values that link the annotation with the text segment they refer to
- *features*: a set of named values. Each indexed feature must have one of the following types:
  - *nominal*: when the permitted values are strings from a limited set
  - *numeric*: floating-point numbers representable in double precision
  - *text*: arbitrary string values
  - *URI*: URIs are used to create links to resources (such as classes or entities) in semantic knowledge bases

When searching for annotations, the user needs to describe their request by providing an annotation *type* and, optionally, one or more *feature constraints*. An annotation query takes the following form: {Type feature1=value1 feature2=value2 ...}.

While the example above uses equality for the feature constraints, other operators are also available. Here is the full list:

**Equality:** Represented by the sign = matches annotations which have the given value for the specified feature. The equality operator is applicable to features of any type.

**Comparison Operators:** Represented by one of the following symbols: <, <=, >, >=, with the usual meaning. These operators can apply to features of type nominal, numeric, or text.

**Regular Expressions:** Can be specified using the syntax REGEX(pattern, flags), where the pattern represents the regular expression sought, and the flags are optional, and can be used to change the way matching is performed. See <http://www.w3.org/TR/xpath-functions/#regex-syntax> for a full specification of the regular expression support. The REGEX operator can only be used for nominal, and text features.

Some example annotation queries are:

{PatentDocument date > 20070000}this searches for all patent documents published from 2007 onwards.<sup>13</sup>

{Reference type = figure}—retrieves all references to figures within the index.

**Sequence Queries and Gaps:** As sequence is the default operator in Mimir, there is no graphical sign for it: simply writing a set of queries one after another will cause a search for sequences of hits, one from each sub-query. For example, the query “the energy level” is actually a sequence query where the first sub-query searches for the word “the”, the second for “energy”, and the last for “level”. This would match occurrences of the exact phrase ‘the energy level’ in the indexed documents. Note that this is different from the standard behaviour of search engines, the majority of which would simply match documents in which all three query terms occur, in whichever order. This type of searching is also supported in Mimir, through the AND operator which is discussed later in this section.

It is sometimes useful to include gaps in a sequence query, that is, to allow arbitrary text fragments (of specified length) to occur in-between the hits from some of the sub-queries. This can be done by using the gap markers “[n]”, or “[m..n]”. These will match a sequence of length *n*, or with a length of between *m* and *n* of arbitrary tokens.

For example the query “the [2] root:time” will match phrases like “the best of times” or “the worst of times”, whereas the query “the [2..10] root:time” would also match “the best use of one’s time” (where the gap consists of six tokens—five words and an apostrophe).

**AND Operator:** The ‘AND’ (also ‘&’) operator can be used to specify queries that should match document segments that include at least one hit from each of the sub-queries. The results returned will always be the shortest document segments that satisfy the query.

**OR Operator:** OR queries are used to search hits that match one of a set of alternative query expressions. This is indicated by using the ‘OR’ (also ‘|’) operator between

---

<sup>13</sup>In general dates are encoded as yyyyymmdd. This encoding allows dates to be treated as numbers, enabling a wide variety of search restrictions.

the sub-queries. A query of the form `Query1 | Query2` will return hits that match either sub-query `Query1` or sub-query `Query2`.

**IN and OVER Operators:** The operators `IN` and `OVER` are used to search for hits of a query that contain, or are contained in the hits of another query. For example:

`Query1 IN Query2` will match all the hits of `Query1` that are contained in a hit of `Query2`.

`Query1 OVER Query2` will match all hits of `Query1` that contain (are overlapping) a hit of `Query2`.

**Repetition Operator:** The `+` operator can be used to match text segments that comprise a sequence of hits from the same sub-query. The length of the sequence is specified through a number (representing the *maximum* number of repetitions) or through two numeric values (representing the *minimum* and *maximum* number of repetitions). For example:

`"to+3"` will match one, two, or three repeated occurrences of the word `to`. The returned hits will be of the form `"to"`, `"to to"`, or `"to to to"`.

`"{Measurement}+2..5"` will match sequences of two, three, four, or five adjacent `Measurement` annotations.

**Grouping:** In the case of complex queries that include multiple sub-queries, parentheses `'(, ')` can be used to group a set of sub-clauses together.

## 15.4 The Patent Annotation Task

The experiments in this paper are based upon three different kinds of patents taken from the MAREC collection<sup>14</sup>: American (USPTO), Japanese (JP) and European (EPO). The reason for choosing multiple data sources is because the three patent types differ in terms of the metadata, formatting, quality, and legal language used. These differences ensure that the approaches we develop can be applied to a wide range of documents, and hopefully to unseen document types with little loss in performance.

The semantic annotation process adds new metadata to the patents (in the form of XML tags). These new metadata fall into two broad categories; wide and deep annotation types. Wide annotations are intended to cover metadata types that apply to patents in general, and do not depend on the specific subject area of the patent (as identified, for example, by its IPC code). Examples of such metadata include document sections and references to cited literature, examples, figures, claims, and other patents. Deep annotations are specific to one or more subject areas and are of interest to specialised patent searchers. The experiments reported here focus upon automatic annotation of measurements (as they are very important for patent professionals) whilst also being very hard to find using keyword search. This is due to the diverse ways in which they can be expressed via natural language.

---

<sup>14</sup><http://ir-facility.net/prototypes/marec/>.



The benefits from the automatic metadata enrichment process are three-fold. Firstly, information extraction (IE) is capable of dealing with variable language patterns and format irregularities much better than text-based regular expressions. For example, references to other patents can be very diverse: U.S. Patent 4,524,128, Korean laid open utility model application No. 1999-007692. Secondly, once the additional metadata have been added to the patent, IE tools can also carry out data normalisation. Again, taking an example from references to figures or similarly claims, expressions such as “Figures 1–3” or “Claims 5–10” imply references not just to the explicitly mentioned figure/claim numbers but also to all those in between. Lastly, by using text mining techniques we are able to extract a significantly wider range of useful information, than could be obtained via keyword search, and provide it as additional XML tags in the patent documents.

The rest of this section details the metadata we currently extract from patents and highlights some of the problems and how these have been overcome.

### ***15.4.1 Section Annotations***

Patent documents are typically quite long, contain multiple required sections, and use highly formalised legal and technical terminology (with the notable exception of literature references and measurements). Different aspects of the patent application are typically presented in a pre-defined set of sections and subsections (e.g. prior art, patent claims, technical problem addressed and effect). Both USPTO and EPO documents have at least three main parts, *the first page* containing bibliographical data and abstract, *the descriptions part*, and *the claims part*.

Automatic section recognition is based upon identifying typical section titles and using them to automatically partition the text. Pre-existing section markup is used, if available. For instance, Bibliographic Data, Abstract and Claims sections tend to be already annotated in patent documents so we use them directly. There are, however, around 20 different sections within most patents<sup>15</sup> and so most sections still need to be detected automatically.

### ***15.4.2 Reference Annotations***

Reference annotations are used for parts of text that refer to either objects in the current document (e.g. figures, tables, etc.) or to other documents (e.g. scientific papers).

A reference annotation consists of two parts; a header indicating the type of reference, and one or more identifiers which typically consist of a mixture of numbers

---

<sup>15</sup>The number of sections within a patent can vary widely from one patent office to another and even, over time, within the same office. Most of the patents we examined during the reported work do, however, contain around twenty sections.

and letters. For example, in *Figure 1 and 2* the header is *Figure* and the identifiers are *1* and *2*. In *U.S. Pat. No. 3,765,999* the header is *U.S. Pat.* and the identifier is *No. 3,765,999*.

Conjunctive phrases mentioning references to two or more objects of the same reference type are tagged initially as one reference annotation, including the conjunction and all punctuation. For example, *Figures 1 and 2; Claims 1–3; Tables 1 to 10* are first annotated as one Reference each, of type Figure, Claim and Table respectively. The normalisation step then separates these into their constituent references; including all implied references (e.g., to Claim 2).

From an IE perspective, some types of references are much simpler to identify than others. For instance, there is little variability in the way patents refer to figures, tables, claims, equations, and examples. References to other patents tend to be slightly more challenging, as they often include the inventors' names, patent date, or even title—in addition to a simple header and identifier. The hardest of all are the references to external sources, such as published papers (see e.g., Hudson & Hay, *Practical Immunology* (Blackwell Scientific Publications, Oxford, UK, 1980), Chap. 8), which tend to be quite long and typically contain many abbreviations and idiosyncratic formatting. We have also observed significant differences between American and European patents in this respect and had to adapt our IE tools to deal with this accordingly.

### 15.4.3 Measurement Annotations

Most measurements comprise a scalar value followed by a unit, e.g.  $2 \times 10^{-7}$  Torr. Furthermore, two scalar values with or without a unit can be contained in an interval. Sometimes there are also accompanying words, such as “less than” or “between” which are important for professional searchers and, therefore, need also to be marked by the IE tools, e.g., “less than about 0.0015 mm”, “ $2 \times 10^5$  to  $2 \times 10^7$  cpm/ml”. Lastly, we also deal with relative measurements, such as percentages and ratios.

The main challenge involved in recognising measurements in patents comes from the large number of measurement units in existence (e.g., units used in physics patents are very different to those used in engineering ones). Another challenge is that some units have single letter abbreviations. These can introduce ambiguities and therefore require a wider context to be considered in order to determine whether a specific sequence of numbers followed by a letter is indeed a measurement. One frequently encountered example of such ambiguities are temperatures, e.g., “1C” where we need to distinguish correct temperature mentions from other cases, such as references to figures, examples, tables, etc. (as in “see Figure 1C”).

**Table 15.2** The SAMIE components listed in runtime order (items in **bold** were developed specifically for SAMIE, other components were customised as needed)

Processing resource	Description
Cleanup	Remove annotations from previous application runs
Import Relevant Markup	Makes relevant markup from the original document available to the rest of the pipeline
<b>Roman Numerals</b>	Annotates Roman numerals which are used for detecting references
<b>Numbers in Words</b>	Recognises numbers written as words and converts them to actual values
Tokeniser	Pattern matcher for detection of words and other lexical items
Sentence splitter	Regular expression-based detection of sentence boundaries
POS tagger	Addition of part of speech (grammatical categories) to tokens
Gazetteer (case sensitive)	Lookup of known domain terms
Gazetteer (case insensitive)	Lookup of known domain terms, with case insensitive matching
<b>Numbers</b>	Find and annotate all remaining numbers
<b>References Transducer</b>	Find and annotate all the references within the documents
<b>Measurement Tagger</b>	Find and annotate all the measurements within the documents

## 15.5 Automatic Patent Annotation

Our approach to the large scale semantic annotation of patent documents is embodied in an information extraction system called SAMIE. This section discusses both SAMIE and the processing infrastructure we have developed to support large scale IE tasks.

### 15.5.1 SAMIE Architecture

SAMIE is provided as a GATE Application Pipeline consisting of a number of independent modules. The modules which make up the application are shown in runtime order in Table 15.2. The pipeline works as follows:

**NLP Infrastructure:** A basic set of NLP components were used to perform a shallow analysis of the input documents; adding simple linguistic features, such as part-of-speech, to the document. These features are added as annotations on the document.

**JAPE Grammars:** Numbers in the documents were mostly identified using the JAPE pattern matching language [5]. Every number which was recognised was augmented by the addition of a ‘value’ feature holding a double representation of the number. JAPE grammars were also employed to detect and annotate sections and references as described in Sects. 15.4.1 and 15.4.2.

**Measurement Tagger:** The measurement tagger is a complicated mix of JAPE rules and Java code that can recognise valid combinations of known units and reduce the units to a form in which they consist only of SI units. This reduction to SI

units then allows measurements of the same dimension (i.e. length) expressed in different ways (e.g. metres, inches, feet . . .) to be indexed, compared against each other and retrieved no matter the unit expressed by the user. The measurement tagger relies on the previous number annotations to remove spurious matches (i.e. a measurement *nearly* always starts with a number).

To enable complex measurement-based queries we have extended the Mimir query language so that `Measurement` annotations support a special synthetic feature, named `spec` which can be used to specify in natural language a measurement value, or a range of values to search for

The values used by the `spec` feature can take one of two forms:

**number unit:** This will match scalar measurements that have the exact specified value,<sup>16</sup> and interval measurements that contain the specified value. For example, ‘23 cm’ or ‘3 inches’.

**number to number unit:** This will match scalar measurements that fall within the specified interval, and interval measurement that overlap with the specified range. For example, ‘2.5 to 15 amperes’. would match all of the following values: ‘3000 mA’, ‘0 to 5 A’, ‘7 to 100 Amperes’, etc.

In either case unit normalisation is performed, so a query expressed in metres can match annotations expressed in inches, or millimetres, etc. For example, all the following represent the same query:

```
{Measurement spec = "3 to 5 metres"}
{Measurement spec = "300 to 500 cm"}
{Measurement spec = "3000 to 5000 mm"}
{Measurement spec = "118 to 197 inches"}17
```

An evaluation of SAMIE [10] has found that the accuracy of the annotations detailed in this sections is comparable to that of human annotators tasked with producing the same metadata. This evaluation gives us the confidence to apply SAMIE to the task of large scale automatic annotation of patents.

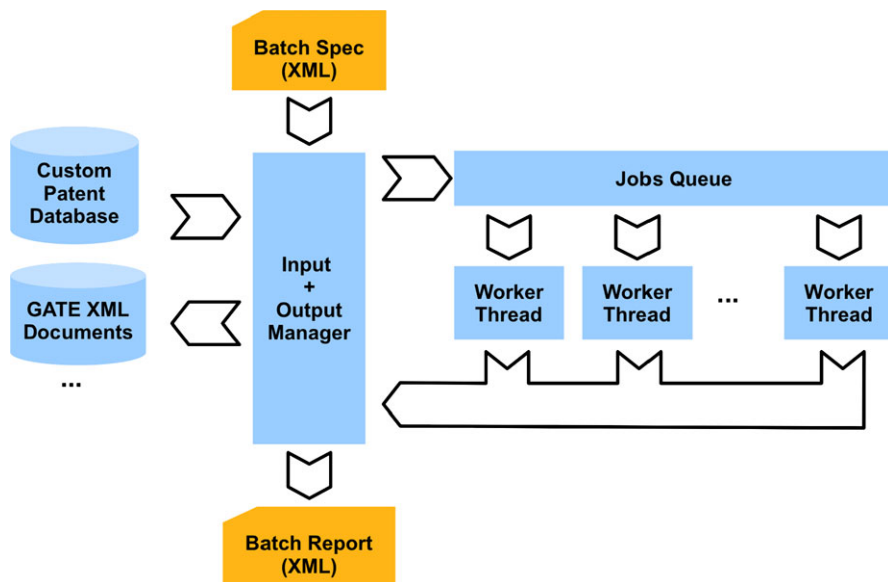
### 15.5.2 Large Scale Annotation with GATE Cloud

One of the main challenges faced in this project is the sheer scale of the task. Patent databases typically contain tens of millions of patents, and hundreds of thousands of new ones are produced each year. Worldwide, millions of new patent applications are submitted yearly.<sup>18</sup> Any application aimed at the IP domain requires a good scalability profile if it is to maintain any credibility.

<sup>16</sup>Within the precision allowed by floating-point arithmetic of double precision.

<sup>17</sup>This query is approximately equal to the others as the two values have been rounded to the nearest whole numbers.

<sup>18</sup>Detailed statistics are available from the World Intellectual Property Organization at <http://www.wipo.int/ipstats/>.



**Fig. 15.1** Overall architecture of GATE Cloud platform

To answer our need for scalability, we have developed GATE Cloud<sup>19</sup>—a platform for parallel semantic annotation of text documents. GATE Cloud is designed as a parallel version of the execution engine found in GATE [7]. It takes a language processing pipeline created using the GATE Developer environment (in this case, the SAMIE application detailed in the previous section), and executes it using a set of parallel threads. The job control is effected through document batches, which are XML files describing outstanding tasks.

A high-level view of the architecture of GATE Cloud is presented in Fig. 15.1. The main elements in the diagram are detailed below:

**Batch Spec:** A batch is a unit of work that needs to be performed. It comprises a list of IDs for the documents that need to be processed, a pointer to the prototype of the processing pipeline that should be used, and configuration data specifying input/output options.

**Input Output Manager:** The I/O manager reads the batch files, parses them, and extracts the IDs for the documents that need to be processed. Its main role is to handle the import/export operations for the patent documents. Internally, GATE Cloud uses GATE Document objects as defined by the GATE Java API; the I/O Manager's job is to create the initial GATE document object for each new document, and to handle the saving of the results at the end of the process. This is also where the integration with various document stores (such as on-disk GATE datastores, or custom patent databases) is handled.

<sup>19</sup><http://gatecloud.net>.

**Jobs Queue:** Each document to be processed represents a job. These are represented as document IDs and are stored in the jobs queue. The queue is accessed in parallel by all the execution threads whenever they become available for work.

**Worker Threads:** A worker thread is a copy of the processing pipeline that manages its own execution thread. Its execution comprises a loop in which it gets the ID for the next available document, it reads the document through the I/O Manager, it executes the processing pipeline over the document, and, finally, it exports the results, again using the facilities provided by the I/O Manager. The number of parallel worker threads is a configuration option for each instance of GATE Cloud, and it depends on the hardware characteristics of the host.

**Batch Report:** The execution of each batch is reflected in a batch report file in XML format. This includes, for each document, whether the execution was successful or some error occurred, and some simple statistics regarding the number of annotations of each type that were produced. Furthermore, once the batch execution completes, details are included regarding the total number of documents processed, how many encountered errors, and the total execution time for the whole batch.

In order to determine the most suitable hardware configuration for running GATE Cloud, we have performed a series of experiments. The main parameters we were trying to estimate were memory requirements, and CPU load, i.e. how many worker threads should be allocated given the number of available CPU cores. Finding the optimal memory allocation is important because low values lead to excessive amounts of CPU time being used for garbage collection, while large values are wasteful. The number of worker threads for a given CPU configuration also needs to be optimised to increase CPU utilisation, while avoiding excessive context switching and locking due to access to shared resources (such as the disk, or network interfaces).

The optimal values will vary depending on the type of documents being processed, and the requirements of the actual processing pipeline used. For each new deployment of GATE Cloud, these parameters should be estimated experimentally. In our particular case, the highest throughput was obtained when each worker thread had 2 GB of RAM allocated, and the number of threads was 1.5 times the number of CPU cores. In this configuration, the *execution speed* was over 1000 documents per hour and per CPU core.

GATE Cloud is designed for parallel execution and it aims at 100% utilisation of a multi-core and/or multi-CPU computer. When combined with an engine for distributed execution of jobs,<sup>20</sup> GATE Cloud can be deployed on large computer farms, or commodity compute clouds. This results in a highly scalable solution for semantic annotation of documents.

GATE Cloud is also intended to run for extended periods of time; conceivably it could even be deployed as a continuously running process. This places some stringent requirements with regard to the robustness of the process, which have influenced the design and implementation. Any errors and exceptions that may occur

---

<sup>20</sup>Such as the Sun Grid Engine (<http://gridengine.sunsource.net/>) or Hadoop (<http://hadoop.apache.org/>).

during processing are trapped and reported, without crashing the entire process. If the GATE Cloud process does crash, for whatever reason (e.g. hardware failure, or power cut), the process can be restarted using exactly the same mechanism as was used to launch it originally. GATE Cloud will automatically identify the previous incomplete run, will parse the partial execution report file to find which documents were already processed successfully, and will resume execution from the point where the previous run stopped.

## 15.6 Multi-Paradigm Patent Search

Whilst some may find the technical details of Mimir interesting, most patent searchers simply wish to know if it will help them to do their job. This section aims to answer that question by showing an example search session over a corpus of 100,000 patents.

The scenario for this example search session involves finding inventions, and their inventors, that make use of transistors.

As with any search engine a good place to start is a keyword-based search.

```
transistor
```

This returns 75,208 hits in the example index—that is the word ‘transistor’ appears 75,208 times within the 100,000 patents. The main problem with this query is that because it matches words, rather than sequences of characters, it does not include any mention of the word ‘transistors’. We could rectify this in one of two ways. In this case, where there are only two variations of the word, we could issue the query `transistor OR transistors`. The problem with this query is that when you have words with more variations, or multiple words where you need to match different tenses, the queries can quickly become unwieldy. A better approach is to use one of the other Mimir string indexes to search on the root form of the word.

```
root:transistor
```

This query now returns 99,742 results. This is a lot of results to search through, and it is likely that most of the results refer to inventions in which transistors only play a minor role. One way to refine the search would be to concentrate on those results which occur within an abstract as this is suggestive of transistors playing an important role in the patent.

```
root:transistor IN {Abstract}
```

Our refined query now returns just 3,053 instances of ‘transistor’ or ‘transistors’ from within the index, but this does not equate to 3,053 different patents.

We can invert the previous query so, that instead of returning all mentions of transistors within abstracts, it instead returns the abstracts which mention transistors.

```
{Abstract} OVER root:transistor
```

The results show that there 1,088 such abstracts within the index. Given the way patents are written they often include multiple abstracts in different languages.<sup>21</sup> We can restrict our query to only focus on abstracts in a given language using the lang feature. For example we could focus on just the 369 abstracts written in French.

```
{Abstract lang=FR} OVER root:transistor
```

Given that we are using the English spelling of transistor (and the index was created using an English part-of-speech tagger) it would, however, make more sense to focus upon just those abstracts written in English.

```
{Abstract lang=EN} OVER root:transistor
```

This query returns 713 abstracts from the 100,000 patent index. Whilst this maybe a small enough number for a team of patent searchers to handle it is likely that refining the search further could be beneficial. One option would be to restrict the search based upon the date of a patent. For example, we could limit the search to only those patents published from 2007 onwards.<sup>22</sup>

```
({Abstract lang=EN} OVER root:transistor)
IN {PatentDocument date > 20070000}
```

This reduces the number of retrieved abstracts to just 321. We can also place an end date on the search in a similar fashion. Restricting the search to just those patents published during 2007 gives us the following query.

```
({Abstract lang=EN} OVER root:transistor)
IN {PatentDocument date > 20070000 date < 20080000}
```

This query retrieves 251 English language abstracts. Whilst this is a useful query (and a reasonable number of results to manually read through), it might be more helpful to start from the title of the inventions rather than the abstracts.<sup>23</sup>

```
{InventionTitle lang=EN}
IN ({PatentDocument date > 20070000 date < 20080000}
OVER ({Abstract lang=EN} OVER root:transistor))
```

As a final example, maybe the aim of this whole search session was to find inventors that you could invite to join an expert pool focusing on transistor-based inventions. We can easily modify the query to retrieve the inventor's instead of the inventions.

```
{Inventor}
IN ({PatentDocument date > 20070000 date < 20080000}
OVER ({Abstract lang=EN} OVER root:transistor))
```

The results from this query shows that there are 2,066 inventors related to the 251 inventions.

---

<sup>21</sup>Whilst this is true for the patents in the MAREC collection, which we used when building this example index, it may not be true for all patents. In fact the structure of patents varies widely which is one reason why effectively searching large patent corpora by hand is difficult.

<sup>22</sup>As previously mentioned, dates are usually encoded as numbers in the form *yyyymmdd*. As such 20070000 is not actually a valid day but does fall between the last day of 2006 and the first day of 2007.

<sup>23</sup>As with abstracts the titles of the inventions are also listed in multiple languages and so a restriction to English is included in the query.



The proceeding examples are of course just a small glimpse into the types of knowledge that can be easily discovered using Mímir. The index used for this example is publicly accessible<sup>24</sup> and we encourage interested readers to try Mímir for themselves.

## 15.7 Discussion and Conclusions

Quoting [8]:

Information retrieval (IR) technology has proliferated in rough proportion to the expansion of knowledge and information as a central factor in economic success. How should end-users choose between them? Three main dimensions condition the choice:

- Volume. The GYM big three web search engines (Google, Yahoo!, Microsoft) deliver sub-second responses to hundreds of millions of queries daily over hundreds of terabytes of data. At the other end of the scale desktop search systems can rely on substantial compute resources relative to a small data set.
- Value. The retrieval of high-value content (typically within corporate intranets or behind pay-for-use turnstiles) is often mission-critical for the business that owns the content. For example the BBC allocates a skilled staff member for eight hours per broadcast hour to index their most important content.
- Cost. Semantic indexing, conceptual search, linked data and so on share with controlled-vocabulary and metadata systems a higher cost of implementation and maintenance than systems based on counting keywords and hyperlinks.

To process web-scale volumes GYM use a combination of one of the oldest and simplest retrieval data structures (an inverted file that relates search terms to documents) and a ranking algorithm whose most important component is derived from the link structure of the web. These techniques work much better than was initially expected, profiting from the vast number of human relevance decisions that are encapsulated in hyperlinks. Problems remain of course: first, there are still many data in which links are not present, and second the familiar problems of ambiguity (*index term synonymy* and *query term polysemy*) can lead to retrieval of irrelevant information and/or failure to retrieve relevant information.

High-value (or low-volume) content retrieval systems address these problems with a variety of semantics-based approaches that attempt to perform conceptual indexing and logical querying. For example, the BBC system cited above indexes using a thesaurus of 100,000 terms that generalise over anticipated search terms. Similarly in the Life Sciences publication databases increasingly use rich terminological resources to support conceptual navigation (MeSH, the Gene Ontology, Snomed, the unified UMLS system, etc.).

An important research theme in recent years has been to ask to what degree can we have our cake and eat it? In other words, how far can the low-volume/high-value methods be extended?

We believe that Mímir makes a contribution to this theme by demonstrating the possibility of scaling up annotation structure indices and combining annotation structure search with full text methods and with conceptual search based on RDF or OWL.<sup>25</sup>

---

<sup>24</sup><http://demos.gate.ac.uk/mimir/patents/gus/search>.

<sup>25</sup><http://www.ontotext.com/owlim/>.

When we began developing Mimir we assumed that we would be able to find an appropriate technology base in a related field such as XML indexing, or database management systems. To this end we convened a workshop in May 2008 on Persisting, Indexing and Querying Multi-Paradigm Text Models (at the IRF in Vienna).<sup>26</sup> A number of researchers working on IR, XML and DBMS were kind enough to participate.<sup>27</sup> It became clear at that point that there was no off-the-shelf solution to our problem (to cut a long story short XML-based techniques were too tree-oriented, and difficult to adapt to the graph structures in which we store annotation data, whereas DBMS techniques are similarly oriented on relational models). Luckily we identified a viable indexing mechanism in the form of MG4J from Sebastiano Vigna,<sup>28</sup> and this forms the core of annotation index management in Mimir.

In this paper we presented the results applied to a use case in patent processing. We also briefly introduced GATE Cloud, our approach to scalability for large annotation tasks. GATE Cloud allows us to take a GATE application and deploy it across machines in a cloud environment allowing the number of documents we can process to be limited only by the machine power available to us.

Together Mimir and GATE Cloud allow us to deliver applications that appear useful in a wide variety of multi-paradigm search contexts.

**Acknowledgements** This work was funded by the Information Retrieval Facility ([ir-facility.org](http://ir-facility.org)). Erik Graf helped us get off the blocks with MG4J and Sebastiano Vigna helped us run the extra mile. Thanks also to all the workshop participants listed above. We are grateful to our reviewers who made salient and constructive contributions.

## References

1. Aswani N, Tablan V, Bontcheva K, Cunningham H (2005) Indexing and querying linguistic metadata and document content. In: Proceedings of fifth international conference on recent advances in natural language processing (RANLP2005), Borovets, Bulgaria
2. Bikel D, Schwartz R, Weischedel R (1999) An algorithm that learns what's in a name. *Mach Learn, Special Issue on Natural Language Learning* 34(1–3)
3. Chomsky N (1999) Profit over people: neoliberalism and global order, 1st edn. Seven Stories Press, New York
4. Cunningham H (2005) Information extraction, automatic. In: *Encyclopedia of language and linguistics*, 2nd edn, pp 665–677
5. Cunningham H, Maynard D, Tablan V (2000) JAPE: a Java annotation patterns engine, 2nd edn. Research Memorandum CS-00-10, Department of Computer Science, University of Sheffield, Nov 2000

---

<sup>26</sup><http://gate.ac.uk/sale/talks/sam/repositories-workshop-agenda.html>.

<sup>27</sup>Gianni Amati (Fondazione Ugo Bordoni/University of Glasgow); Mike Baycroft (Fairview Research); Norbert Fuhr (University of Essen-Duisburg); Eric Graf (University of Glasgow); Atanas Kiryakov (Ontotext); Borislav Popov (Ontotext); Ralf Schenkel (MPG); John Tait (IRF); Arjen de Vries (ACM/CWI); Francisco Webber (Matrixware/IRF); Valentin Tablan (University of Sheffield); Kalina Bontcheva (University of Sheffield); Hamish Cunningham (University of Sheffield).

<sup>28</sup><http://mg4j.dsi.unimi.it/>.

6. Cunningham H, Maynard D, Bontcheva K, Tablan V, Dimitrov M, Dowman M, Aswani N, Roberts I, Li Y, Funk A (2000) Developing language processing components with GATE Version 6.0 (a user guide). <http://gate.ac.uk/>
7. Cunningham H, Maynard D, Bontcheva K, Tablan V (2002) GATE: a framework and graphical development environment for robust NLP tools and applications. In: Proceedings of the 40th anniversary meeting of the association for computational linguistics (ACL'02)
8. Cunningham H, Hanbury A, R uger S (2010) Scaling up high-value retrieval to medium-volume data. In: Cunningham H, Hanbury A, R uger S (eds) Advances in multidisciplinary retrieval (the 1st information retrieval facility conference). LNCS, vol 6107. Vienna, Austria, May 2010. Springer, Berlin
9. Day D, Robinson P, Vilain M, Yeh A (1998) MITRE: description of the *Alembic* system used for MUC-7. In: Proceedings of the seventh message understanding conference (MUC-7)
10. Greenwood MA, Cunningham H, Aswani N, Roberts I, Tablan V (2010) GATE Mimir: philosophy, development, deployment and evaluation. Research Memorandum CS-10-05, Department of Computer Science, University of Sheffield
11. Hull D, Ait-Mokhatar S, Chuat M, Eisele A, Gaussier E, Grefenstette G, Isabelle P, Samuelsen C, Segond F (2001) Language technologies and patent search and classification. World Pat Inf 23:265–268
12. Li Y, Bontcheva K, Cunningham H (2005) SVM based learning system for information extraction. In: Winkler MNJ, Lawrence N (eds) Deterministic and statistical methods in machine learning. LNAI, vol. 3635. Springer, Berlin, pp 319–339
13. Maynard D, Tablan V, Ursu C, Cunningham H, Wilks Y (2001) Named entity recognition from diverse text types. In: Recent advances in natural language processing 2001 conference, Tzigov Chark, Bulgaria, pp 257–274
14. Maynard D, Bontcheva K, Cunningham H (2003) Towards a semantic extraction of named entities. In: Recent advances in natural language processing, Bulgaria
15. Wanner L, Baeza-Yates R, Brugmann S, Codina J, Diallo B, Escorsa E, Giereth M, Kompatsiaris Y, Papadopoulos S, Pianta E, Piella G, Puhlmann I, Rao G, Rotard M, Schoester P, Serafini L, Zervaki V (2008) Towards content-oriented patent document processing. World Pat Inf 30(1):21–33

# Chapter 16

## Intelligent Information Access from Scientific Papers

**Ted Briscoe, Karl Harrison, Andrew Naish, Andy Parker, Marek Rei, Advaith Siddharthan, David Sinclair, Mark Slater, and Rebecca Watson**

**Abstract** We describe a novel search engine for scientific literature. The system allows for sentence-level search starting from portable document format (PDF) files, and integrates text and image search, thus, for example, facilitating the retrieval of information present in tables and figures using both image and caption content. In addition, the system allows the user to generate in an intuitive manner complex queries for search terms that are related through particular grammatical (and thus implicitly semantic) relations. Grid processing techniques are used to parallelise the

---

T. Briscoe (✉) · K. Harrison · A. Parker · M. Rei · M. Slater  
University of Cambridge, Cambridge, UK  
e-mail: [Ted.Briscoe@cl.cam.ac.uk](mailto:Ted.Briscoe@cl.cam.ac.uk)

K. Harrison  
e-mail: [Harrison@hep.phy.cam.ac.uk](mailto:Harrison@hep.phy.cam.ac.uk)

A. Parker  
e-mail: [Parker@hep.phy.cam.ac.uk](mailto:Parker@hep.phy.cam.ac.uk)

M. Rei  
e-mail: [Marek.Rei@hep.phy.cam.ac.uk](mailto:Marek.Rei@hep.phy.cam.ac.uk)

M. Slater  
e-mail: [Slater@hep.phy.cam.ac.uk](mailto:Slater@hep.phy.cam.ac.uk)

T. Briscoe · R. Watson  
iLexIR Ltd, Cambridge, UK

R. Watson  
e-mail: [Bec.Watson@gmail.com](mailto:Bec.Watson@gmail.com)

A. Naish · A. Parker · D. Sinclair  
Camtology Ltd, Cambridge, UK

A. Naish  
e-mail: [A.Naish@gmail.com](mailto:A.Naish@gmail.com)

D. Sinclair  
e-mail: [David.Sinclair@imense.co.uk](mailto:David.Sinclair@imense.co.uk)

A. Siddharthan  
University of Aberdeen, Aberdeen, UK  
e-mail: [Advaith@abdn.ac.uk](mailto:Advaith@abdn.ac.uk)

analysis of large numbers of scientific papers. We are currently conducting user evaluations, but here we report some preliminary evaluation and comparison with Google Scholar, demonstrating the potential utility of the novel features. Finally, we discuss future work and the potential and complementarity of the system for patent search.

## 16.1 Introduction

Scientific, technological, engineering and medical (STEM) research is entering the so-called 4th Paradigm of “data-intensive scientific discovery” in which advanced data mining and pattern discovery techniques need to be applied to vast datasets in order to drive further discoveries [13]. A key component of this process is efficient search and exploitation of the huge repository of information that only exists in textual or visual form within the “bibliome”, which itself continues to grow exponentially.

Today’s computationally driven research methods have outgrown traditional methods of searching for scientific data creating a significant, widespread and unfulfilled need for advanced search and information extraction. Our system fully integrates text and content-based image processing in order to create a unique solution to fine-grained search and information extraction for scientific papers. In this paper, we describe the current version of our system demonstrator focussing on its search capabilities.

We have developed a prototype search and information extraction system, which is currently undergoing usability testing with the curation team for FlyBase, a \$1 m/year NIH-funded curated database covering the functional genomics of the fruit fly. To provide a scalable solution capable, in principle, of analysing the entire STEM bibliome of around 20 m electronic journal and conference papers, we have developed a distributable and robust system that can be used with a grid of computers running distributed job management software.

This system has been deployed and tested using a subset of the resources provided by the UK Grid for Particle Physics [3], part of the worldwide grid assembled for the analysis of the petabyte-scale data volumes to be recorded each year by experiments at the Large Hadron Collider in Geneva. To build the current demonstrator we processed around 15k papers requiring about 8k hours of CPU time in about 3 days with up to 100 jobs running in parallel. A distributed spider for finding and collecting open access portable document format (PDF) versions of papers has also been developed. This has been run concurrently on over 2k cores, and has been used to retrieve over 1m subject-specific papers from a variety of STEM fields to date. However, the demonstrator, as discussed below, indexes about 10k papers on the functional genomics of the fruit fly.

## 16.2 Functionality

Our search and extraction engine is the first to integrate a full structural analysis of a scientific paper in PDF identifying headings, sections, captions and associated figures, citations and references with a sentence-by-sentence grammatical analysis of the text and direct content-based visual search over figures. Combining these capabilities allows us to transform paper search from keyword-based paper retrieval, where the end result is a set of putatively relevant PDF files which need to be read, to information search and extraction, based on the ability to interactively specify a rich variety of linguistic patterns which return sentences in specific document locales and which combine text with image-based constraints—for instance:

all sentences in figure captions which contain any gene name as the theme of *express* where the figure is a picture of an eye

The system allows the user to build up such complex queries quickly though an intuitive process of query refinement.

Figures often convey information crucial to the understanding of the content of a paper and are typically not available to search. Our search engine integrates text search to the figure and caption level with the ability to re-rank search returns on the basis of visual similarity to a chosen archetype (ambiguities in textual relevance are often resolved by visual appearance). Figure 16.1 provides a compact overview of the search functionality supported by the demonstrator.

Interactively, constructing and running such complex queries takes a few seconds in our intuitive user interface, and allows the user to quickly browse and then aggregate information across the entire collection of papers indexed by the system. For instance, saving the search result from the example above would yield a computer-readable list of gene names involved in eye development in less than a second on a standard 64-bit machine indexing around 10k papers. With existing web portals and keyword-based selection of PDF files (for example, Google Scholar, ScienceDirect, Zotero or Mendeley), a query like this would typically take many hours to execute, requiring each PDF file returned to be opened and read in a PDF viewer, and cut and paste to extract relevant gene names.

The only other current solution would require expensive customisation of a text mining/information extraction system by IT professionals using licensed software (such as that provided by Ariadne Genomics, Temis or Linguamatics). This option is only available to a tiny minority of researchers working for large well-funded corporations.

## 16.3 Summary of Technology

### 16.3.1 PDF to SciXML

PDF was developed to represent a document in a manner designed to facilitate printing. In short, it provides information on font and position for textual and

**1 search by keyword**

**2 identify interesting sentences**

**3 find similar sentences based on grammatical structure**

**4 results match form of query, not just content**

**5 browse by MeSH term**

**6 search by caption**

**7 view images and tables directly in the browser**

**8 easily find similar images based on content alone**

**9 results appear on world map**

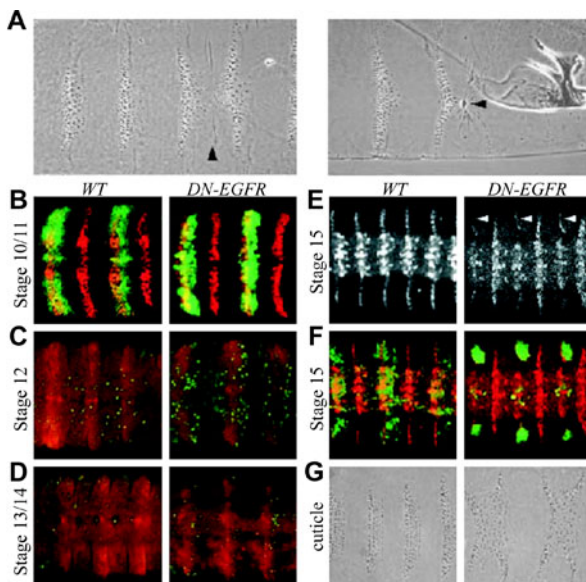
Fig. 16.1 Screenshots showing functionality of our demonstrator

graphical units. To enable information retrieval and extraction, we need to convert this ubiquitous typographic representation into a logical one that reflects the structure of scientific documents ([10]). We use an XML schema called SciXML ([14]) that we extend to include images. We linearise the textual elements in the PDF, representing these as <div> elements in XML and classify these divisions as {Title|Author|Affiliation|Abstract|

Footnote|Caption| Heading|Citation|References|Text} in a constraint satisfaction framework.

In addition, we identify all graphics in the PDF, including lines and images. We then identify tables by looking for specific patterns of text and lines. A bounding box is identified for a table and an image is generated that overlays the text on the lines. Similarly we overlay text onto images that have been identified and define bounding boxes for figures. This representation allows us to retrieve figures and tables that consist of text and graphics. Once bounding boxes for tables or figures have been identified, we identify a one-one association between captions and boxes that minimises the total distance between captions and their associated figures or tables. The image is then referenced from the caption using a “SRC” attribute; for example, in (abbreviated for space constraints):

<CAPTION SRC="FBrf0174566\_fig\_6\_o.png">  
 <b>Fig. 6. </b> Phenotypic analysis of denticle belt fusions during embryogenesis. (A) The denticle belt fusion phenotype resulted in folds around the surrounding fused... ..(G) ...the only cuticle phenotype of the DN-EGFR-expressing embryos was strong denticle belt fusions in alternating parasegments (<i>paired</i> domains).</CAPTION>



Note how informative the caption is, and the value of being able to search this caption in conjunction with the corresponding image (also shown above).



## 16.3.2 Natural Language Processing

Every sentence or smaller textual unit, including those in abstracts, titles and captions, is run through our named-entity recogniser (NER) and syntactic parser. The output of these systems is then indexed, enabling more precise search.

### 16.3.2.1 Named Entity Recognition

NER in the biomedical domain was implemented as described in [15]. Gene Mention tagging was performed using a Conditional Random Fields model (using the MALLET toolkit [11]) and syntactic parsing (using RASP [2], using features derived from grammatical relations to augment part-of-speech (PoS) tagging. We also use a probabilistic model for resolution of non-pronominal anaphora in biomedical texts. The model focuses on biomedical entities and seeks to find the antecedents of anaphoric expressions, both coreferent and associative ones, and also to identify discourse-new expressions [5], and we deploy a reference parsing and citation linking module during the processing pipeline. The combination of these modules allows us to identify and distinguish mentions of author names, gene names, and gene products or components such as protein names, DNA sequence references, and so forth.

Both the NER and anaphora resolution modules of our processing pipeline are domain-specific. However, both are weakly supervised and rely on extant ontologies or domain information, such as the gene names recorded in FlyBase, to generate training data and/or dictionaries. Therefore, these components are extensible to further scientific subfields for which similar ontologies and resources can be found.

### 16.3.2.2 Parsing

The RASP (Robust Accurate Statistical Parsing [2]) toolkit is used for sentence boundary detection, tokenisation, PoS tagging, morphological analysis and finding grammatical relations (GR) between words in the text. GRs are triplets consisting of a relation-type and two arguments and also encode morphology, word position and part-of-speech; for example, parsing “John likes Mary.” gives us a subject relation and a direct object relation:

```
([ncsubj] |like+s:2_VVZ| |John:1_NP1|)
([dobj] |like+s:2_VVZ| |Mary:3_NP1|)
```

Representing a parse as a set of flat triplets allows us to index on grammatical relations [1], thus enabling more complex relational queries than is standard in scientific search engines.

The RASP system is relatively domain-independent compared to alternative statistical parsers. Lexical information is only used within the PoS tagger which also integrates a sophisticated unknown word handling module. The parser operates on

PoS tag sequences and ranks alternative parses using structural information drawn from balanced training data. Nevertheless to improve handling of the large proportion of unknown words, we use the predictions of the NER module to retag names with the correct PoS tag in cases where the tagger chooses an alternative, and we save the top five highest-ranked parses for indexing to improve recall in cases where the preferred parse is not correct.

### 16.3.3 Image Processing

We build a low-dimensional feature vector to summarise the content of each extracted image. Colour and intensity histograms are encoded in a short bit string which describes the image globally; this is concatenated with a description of the image derived from a wavelet decomposition [9] that captures finer-scale edge information. Efficient similar image search is achieved by projecting these feature vectors onto a small number of randomly generated hyperplanes and using the signs of the projections as a key for locality-sensitive hashing [6].

Thus our current image similarity search is based on unsupervised clustering with some tuning of feature weights to achieve useful results in this domain. In the near future we will add supervised classifiers capable of recognising common subclasses of image occurring in papers, such as graphs, plots, photographs, etc., based on training data derived automatically via captions unambiguously identifying the accompanying image type.

### 16.3.4 Indexing and Search

We use Lucene [7] for indexing and retrieving sentences and images. Lucene is an open source indexing and information retrieval library that has been shown to scale up efficiently and handle large numbers of queries. We index using fields derived from word-lemmas, grammatical relations and named entities. At the same time, these complex representations are hidden from the user, who, as a first step, performs a simple keyword search; for example *express Vnd*. This returns all sentences that contain the words *express* and *Vnd* (search is on lemmatised words, so morphological variants of *express* will be retrieved). Different colours represent different types of biological entities and processes (green represents a gene), and blue words show the entered search terms in the result sentences an example sentence retrieved for the above query follows:

It is possible that like **ac**, **sc** and **l'sc**, **vnd** is expressed initially in cell clusters and then restricted to single cells.

Next, the user can select specific words in the returned sentences to indirectly specify a relation. Clicking on a word will select it, indicated by underlining of the word. In the example above, the words *vnd* and *expressed* have been selected by

the user. This creates a new query that returns sentences where *vnd* is the subject of *express* and the clause is in passive voice. This retrieval is based on a sophisticated grammatical analysis of the text, and can retrieve sentences where the words in the relation are far apart; an example of a sentence retrieved for the refined query is shown below:

First, **vnd** might be spatially regulated in a manner similar to **ac** and **sc** and selectively *expressed* in these clusters.

Once a user is confident that a ground pattern of this type is retrieving relations of interest appropriately, it is possible to ‘wildcard’ an argument of a predicate or abstract from a specific member of a semantic class, such as the gene *Vnd* to the entire class, in this case of genes. Figure 16.1 (step 3) shows a screenshot of the interface supporting this functionality

The current demonstrator offers two further functionalities. The user can browse the MeSH (Medical Subject Headings) ontology and retrieve papers relevant to a MeSH term. Also, for both search and MeSH browsing, retrieved papers are plotted on a world map; this is done by converting the affiliation of the first author into geospatial coordinates. The user can then directly access papers from a particular research group indexed with a specific MeSH term.

## 16.4 Evaluation

The demonstrator is currently undergoing user trials with members of the FlyBase curation team. They are faced with an increasing number of papers that they have identified as potentially curatable and downloaded on the basis of keyword search. The process of deciding whether a paper should be fully curated (approximately a person/day of effort), lightly curated recording, for example, genes mentioned, or ignored is itself time consuming and currently done by uploading a PDF to a viewer and/or printing it, and then reading it.

The system potentially speeds up this process by allowing a collection of papers to be searched at the sentence level for key phrases that indicate relevant information. For example, predicates such as *characteriz/se* with gene names as objects often indicate new information about a gene, whilst assignment of a mnemonic name to a sequenced gene, denoted by a numerical identifier prefixed with *CG*, is a good clue that a paper contains the first significant investigation of that gene. The ability to define patterns in the interface that find such characterisation or naming events from the text, means that, in principle, fully curatable papers can be identified much more quickly.

Although it is too early to report on these usability experiments, we have conducted preliminary exploration of some common types of searches using intrinsic evaluation methods common in Information Retrieval, such as the (Mean) Average Precision measure. This is appropriate when we are evaluating a system that ranks sentences according to a given query where we want to measure the degree to which

relevant sentences are ranked higher than irrelevant sentences and all relevant sentences appear in the ranking. A single query version of average precision is defined by

$$\frac{\sum_{r=1}^N (Prec(r) \times TP?(r))}{TruePositives + FalseNegatives}, \quad (16.1)$$

where  $N$  is the number of sentences returned by the system,  $r$  is the rank of the sentence, and  $TP?$  returns one (zero) if the  $r$ th sentence is (not) a true positive and  $Prec(ision)$  is defined as

$$\frac{TruePositives}{TruePositives + FalsePositives} \quad (16.2)$$

so a score of one entails perfect recall and ranking [16].

We start by considering a relatively simple goal like “find all sentences which discuss Adh expression in fruit flies” where Adh is a gene name and we are interested in expression events with Adh as theme. As illustrated in Sect. 16.3.4, keyword search can be refined to enforce the appropriate semantic relation between the gene name and some form of the predicate *express*, and near synonyms such as *overexpress* if desired. The goal then is to retrieve sentences containing phrases like (a), (b) or (c) below, but not (d).

- (a) ...express Adh...
- (b) ...expression of Adh...
- (c) Adh is one of the most highly expressed genes...
- (d) Adf-1 is an activator of Adh that was subsequently shown to control expression of several *Drosophila* genes...

Our system allows the user to achieve this by constructing a (disjunctive) set of queries which define various appropriate grammatical patterns, Note that standard IR and search engine refinements like string search or operators like NEAR cannot achieve the same effect. The former achieving high precision but low recall, the latter achieving a better approximate ranking, but not directly enforcing grammatical/semantic constraints.

To achieve this goal using Google Scholar (or any other document-level search system, such as those offered by the major scientific publishers, academic associations, etc.) a sophisticated user might construct the following query:

(Adh OR alcohol dehydrogenase OR CG32954) NEAR (expression OR express OR overexpress) AND *Drosophila*

This yielded about 15k papers together with header and text snippets (in July 2010). Using the headers and snippets, the user now has to decide whether to save a PDF for further investigation or not. The information available before downloading and opening the paper in a PDF viewer is sometimes adequate to accept or reject a paper, but also often unclear. For example, the snippet in (a) below clearly shows this paper contains a relevant sentence; that in (b) strongly suggests the paper contains no relevant sentence, but that in (c) is unclear because the first snippet has been truncated after *the* so the critical information is missing.

- (a) Identification of cisregulatory elements required for larval expression of the *Drosophila melanogaster* alcohol dehydrogenase gene. . . .
- (b) Hypomorphic and hypermorphic mutations affecting the expression of Hairless. . . . The genetics of a small autosomal region of *Drosophila melanogaster*, including the structural gene for alcohol dehydrogenase.
- (c) The Molecular Evolution of the Alcohol Dehydrogenase and Alcohol Dehydrogenase-related Genes in the . . . The DNA sequences of the A&z genes of three members of the *Drosophila melanogaster* species . . .

Furthermore, the ranking of papers given by Google Scholar does not ensure that clearly relevant snippets occur before unclear or irrelevant ones, as ranking appears to be based on a combination of the frequency of keyword occurrences through the paper and on keyword density within snippets. For example, (b) above occurs before (c), whilst the 50th page of results still contains three (out of ten) papers with clearly relevant snippets, and the 99th page one clearly relevant snippet. Indeed, after the first few pages where most snippets are clearly relevant, the ranking ‘flattens’ so that most pages sampled throughout the set returned contain one or two clearly relevant snippets.

We estimate that a comprehensive search of papers with relevant snippets would involve downloading and viewing about 1K papers, though even then there would be little hope of achieving full recall, given the indeterminacy of a significant number of headers and snippets. For each paper downloaded, a PDF viewer’s Find feature can be used to quickly move to potentially relevant sentences. We sampled 10 papers with relevant snippets and found that in general *express* was the more restrictive keyword. On average, we found about 100 matching sentences of which about 10 exhibited the relevant relationship, whilst it took about 10 minutes per paper to identify these sentences. A conservative estimate of the time taken to identify the entire set of relevant sentences in papers clearly identified as relevant by Google Scholar would be about one month. The average precision of this approach—assuming that relevant sentences within papers are uniformly distributed, factoring in snippet identification, but assuming full recall via clearly relevant snippets—would be about 0.1 over the first 30 or so papers and about 0.001 over the full set. In a sense this analysis is unfair as Google Scholar is designed to be a paper retrieval system. Nevertheless, it is probably the best generally available tool for the task today, as the snippet information surpasses anything provided by other scientific paper search sites, such as Elsevier’s ScienceDirect, and its coverage of the literature is unrivalled.

To estimate performance in our demonstrator we used the Lucene command-line query language back-end to retrieve all sentences which contained a form of *express* or one of its near synonyms and *Adh* or one of its synonyms. We then manually classified this set of sentences into those which were relevant or not, and used this gold standard to compute average precision scores for four variant queries. Query 1 simply used the ranking obtained searching for *Adh* and *express* in the same sentence, query 2 required some path of grammatical relations linking these two keywords, query 3 added synonyms for each keyword, and query 4 enforced some path of grammatical relations between each set of synonyms and scored the sentences for

**Table 16.1** Average Precision for *Adh* as theme of *express*

Query	1	2	3	4
	0.735	0.758	0.855	0.933

**Table 16.2** Average Precision for *CG* naming events

Query	1	2	3	4	5
	0.116	0.461	0.552	0.512	0.562

ranking according to the length of this path to favour shorter paths. The average precision for each of these queries is given in Table 16.1. Gains in precision of several orders of magnitude are made over using Google Scholar and a PDF viewer, simply by supporting (Boolean) keyword search over sentences and returning these rather than PDFs. However, grammatical constraints also yield a significant improvement in the overall ranking obtained, effectively ensuring that, for the first two pages of results returned, all sentences are relevant.

So far, we have only considered searches involving ground terms, but the system allows search via patterns over semantic/named entity classes or partially wildcarded terms. As mentioned above, curators would like to find papers that contain naming events involving *CG* prefixed identifiers, as these are a useful clue that a paper should be fully curated with respect to the named gene. We used the Lucene query language to find sentences containing variants of the predicate *name* (*X Y*) and synonyms like *call* (*X Y*), *refer* (*to X as Y*), etc along with any lemma matching *CG\** and then manually classified the resulting set to identify relevant sentences containing a naming event between the *CG* identifier and a gene name. We then used this gold standard to compute average precision for five variant queries. Query 1 simply searched for sentences containing *CG\** and a variant of *name*, query 2 added synonyms of *name* as above, query 3 disjunctively specified a set of known patterns that picked out grammatical constructions likely to specify a naming relation, like ‘*CGID referred to as GENE*’ or ‘*CGID (GENE)*’, query 4 allowed any path of grammatical relations between the *CG* identifier and a naming predicate scored by length, and query 5 combined the specific grammatical patterns (query 3) and the general path constraint (query 4). The average precision for each of these queries is given in Table 16.2. In the case of this more complex relational query between classes of terms, overall performance is poorer but the differential advantage of enforcing grammatical constraints is also much greater in this case than a simple requirement for co-occurrence of terms within a sentence.

The current user interface doesn’t support the general path constraint on grammatical relations. Therefore, curators need to disjunctively specify a range of grammatical patterns and collate the results of each of these manually. We plan to redesign the system to support automatic expansion of queries to add semantically equivalent grammatical patterns and to enforce the path constraint by default in refined searches specifying any grammatical constraint. For instance, returning to the example in Sect. 16.3.4, a user who selects *express* and *Vnd* in a sentence where

*Vnd* is the subject of the passive verb group *is expressed* would automatically be shown further sentences in which *Vnd* is object of an active or nominalised form of the verb, such as *expressed Vnd* or *expression of Vnd*, and sentences in which any path of grammatical relations between a form of *express* and *Vnd* is found, such as *expression of ac and sc often with Vnd*, would be returned, albeit with lower ranking.

The ranking of search results yielded by complex queries with multiple constraints on images and text is sometimes unintuitive, as are the results of similarity-based image search. We are adding classification to the image search, exploiting caption information to gather labelled training data, so that results hopefully will be less arbitrary than those sometimes achieved by unsupervised clustering. We are moving to a faceted, Boolean model of query constraint integration, so that scoring and ranking of results will play a less central role in “navigation” towards a satisfactory query formulation.

Nevertheless, even using the current interface it is possible to identify sets of papers, using queries of this type, for full curation with satisfactory recall and good enough precision. This process takes less than an hour rather than the weeks required to achieve similar ends using other widely available scientific paper search systems.

## 16.5 Related Work

The current system draws insights from existing work in information retrieval, information extraction, and biomedical text mining. For instance, other researchers have recognised that document retrieval via images and figures, or their direct retrieval, is useful with scientific papers [4], though we are not aware of any work which integrates the search of images and text via an interface that supports iterated refinement of multimodal search facets used to jointly rank retrieved text and images. Work in information extraction from biomedical text has demonstrated the value of syntactic parsing and named entity recognition, for example, for the extraction of protein-protein interactions [12]. However, typically such information extraction systems need significant customisation for each such subtask. In information retrieval, work on the TREC Genomics tasks and datasets has demonstrated the value of, for instance, novel query expansion [17] and ranking [8] techniques. However, this body of work is focussed on document / passage retrieval and classification, and it is not clear that the insights gained or techniques developed are directly applicable to the more generic search and information extraction scenario considered here.

## 16.6 Conclusions and Further Work

To our knowledge, this is the first time that content-based image and advanced text processing have been fully integrated to provide fine-grained and multimodally faceted search over scientific papers. Our preliminary experiments suggest that the

resulting system has the potential to greatly improve search and information extraction with complex documents. In order to develop the system in a scalable and relatively domain-independent fashion, we have utilised the grid and distributed processing to spider and annotate papers, and weakly supervised machine learning methods or domain-independent modules in the annotation process. Our annotation pipeline is the first developed which is able to preprocess a PDF, identify the internal structure, and represent the result in a manner which supports application of state-of-the-art image and text processing techniques.

Nevertheless, there is much work to be done before all of our aims are achieved. Firstly, we need to demonstrate that the weakly supervised NER and anaphora resolution modules can be ported effectively to new (sub-)domains or that they can be replaced without serious loss of search performance by unsupervised techniques. Secondly, we need to evaluate the user interface with a wider group of potential users and to explore and develop its effectiveness for other fields, such as computer science, which differ from genetics in terms of the likely focus of searches. Thirdly, we need to extend system functionality and the interface to support information extraction. This will require the ability to save and reapply complex queries once they have been developed incrementally and interactively to a point where the user is satisfied with their performance. Where these (relational) queries match classes of terms, it would also be useful to be able to save the lists of ground terms that match in a computable-readable format and also to re-use such lists during the formulation of further queries.

The system is potentially relevant to patent search professionals for several reasons. Firstly, we believe that the techniques we have developed for search and information extraction from scientific papers are broadly applicable to any collection of relatively complex documents containing technical terminology, images, and internal structure, such as patents. In addition, our current demonstrator also supports access to papers via the MeSH ontology, and this could be straightforwardly extended to support access to patents via any of the ontologies developed to support patent search. Secondly, there are many similarities between patent and scientific paper search which demarcate both from general web search. Both often involve fine-grained and comprehensive search for information rather than keyword-based access to a document or page ranked by popularity or frequency of keywords. And both are conducted by professionals willing to develop 'advanced search' expertise whose search sessions typically last hours rather than minutes. Finally, patent searchers are frequently interested in prior art, and prior art can potentially be found in the scientific bibliome. In the longer run, combined search over both patents and scientific papers using the same interface and search tools would be very valuable.

**Acknowledgements** This work was supported in part by a BBSRC e-Science programme grant to the University of Cambridge (FlySlip), and a STFC miniPIPSS grant to the University of Cambridge and iLexIR Ltd (Scalable and Robust Grid-based Text Mining of Scientific Papers). This chapter is an extended version of one which appeared in the proceedings of the annual North American Association for Computational Linguistics conference proceedings, demonstration session, in June 2010.



## References

1. Atterer M, Schutze H (2008) An inverted index for storing and retrieving grammatical dependencies. In: Proceedings of the 6th international conference on language resources and evaluation, Marrakech, Morocco
2. Briscoe T, Carroll J, Watson R (2006) The second release of the rasp system. In: Proceedings of the COLING/ACL 2006, Sydney, Australia
3. Britton D, Cass AJ, Clarke PEL, Coles J, Colling DJ, Doyle AT, Geddes NI, Gordon JC, Jones RWL, Kelsey DP et al (2009) GridPP: the UK grid for particle physics. *Philos Trans A* 367(1897):2447
4. Eggel I, Müller H (2010) Indexing the medical open access literature for textual and content-based visual retrieval. *Stud Health Technol Inf* 160(2):1277–1281
5. Gasperin C, Briscoe T (2008) Statistical anaphora resolution in biomedical texts. In: Proceedings of the 22nd international conference on computational linguistics, vol 1, pp 257–264
6. Gionis A, Indyk P, Motwani R (1999) Similarity search in high dimensions via hashing. In: Proc 25th internat conf on very large data bases
7. Goetz B (2002) The Lucene search engine: powerful, flexible, and free. *Javaworld* <http://www.javaworld.com/javaworld/jw-09-2000/jw-0915-lucene.html>
8. Huang X, Hu Q (2009) A Bayesian learning approach to promoting diversity in ranking for biomedical information retrieval. In: Proceedings of SIGIR 2009, Boston, MA. ACM 978-1-60558-483-6/09/07
9. Jacobs CE, Finkelstein A, Salesin DH (1995) Fast multiresolution image querying. In: Proceedings of the 22nd annual conference on computer graphics and interactive techniques. ACM, New York, pp 277–286
10. Lewin I, Hollingsworth B, Tidhar D (2005) Retrieving hierarchical text structure from typeset scientific articles: a prerequisite for e-science text mining. In: Proceedings of the 4th UK E-science all hands conference, Nottingham, UK, pp 267–273
11. McCallum AK (2002) Mallet: A machine learning for language toolkit. <http://mallet.cs.umass.edu>
12. Saetre R, Matsuzaki T, Miyao Y, Sagae K, Tsujii J (2009) Evaluating contributions of natural language parsers to protein-protein interaction extraction. *Bioinformatics* 25(3):394–400
13. Tansley S, Hey T, Tolle K (2009) The fourth paradigm: data-intensive scientific discovery. Microsoft Research, Redmond
14. Teufel S, Carletta J, Moens M (1999) An annotation scheme for discourse-level argumentation in research articles. In: Proceedings of the 9th conference of the European chapter of the association for computational linguistics (EACL'99), pp 110–117
15. Vlachos A (2007) Tackling the BioCreative2 gene mention task with conditional random fields and syntactic parsing. In: Proceedings of the second BioCreative challenge evaluation workshop
16. Voorhees E, Harman K (1999). The seventh text retrieval conference (TREC-7). NIST
17. Wang S, Hauskrecht M (2010) Effective query expansion with the resistance distance based term similarity metric. In: Proceedings of SIGIR 2010, Geneva, Switzerland. ACM 978-1-60558-896-4/10/07

# Chapter 17

## Representation and Searching of Chemical-Structure Information in Patents

John D. Holliday and Peter Willett

**Abstract** This chapter describes the techniques that are used to represent and to search for molecular structures in chemical patents. There are two types of structures: specific structures that describe individual molecules; and generic structures that describe sets of structurally related molecules. Methods for representing and searching specific structures have been well established for many years, and the techniques are also applicable, albeit with substantial modification, to the processing of generic structures.

### 17.1 Introduction

Patents are a key information resource for all types of industry, but this is particularly the case in the pharmaceutical and agrochemical industries. The main focus of these industries is to identify novel chemical molecules that exhibit useful biological activities, e.g., reducing an individual's cholesterol level or killing the insect pest of a crop [1, 5]. Chemical patents hence need to contain not just the textual information that one would find in any type of patent, but also information about the chemical molecules of interest. These can, of course, be described by their chemical names or images, but these provide only limited searching facilities that are not sufficient to meet the requirements of modern industrial research and development. Instead, specialised types of representation and search algorithm have had to be developed to provide efficient and effective access to the structural information contained in patents. These techniques are an important component of what has come to be called *chemoinformatics* [34], i.e., “the application of informatics methods to solve chemical problems” [15].

Two types of molecular information are encountered in chemical patents. A patent may be based on just a single specific molecule, in which case the techniques that have been developed in chemoinformatics over many years may be applied, as discussed below. However, the majority of chemical patents discuss not single molecules, but entire classes of structurally related molecules, with these

---

J.D. Holliday · P. Willett

Information School, University of Sheffield, 211 Portobello Street, Sheffield S1 4DP, UK

classes being described by a *generic*, or *Markush*, structure. A single generic structure can represent many thousands, or even a potentially infinite number, of individual molecules, and the representational and searching techniques required are accordingly far more complex than those commonly encountered in chemoinformatics systems. In this paper, we provide an overview of the techniques that are used to handle both specific and generic chemical structures. The reader is referred to the standard texts by Leach and Gillet [23] and by Gasteiger and Engel [16] for further details of the techniques described below; these books also provide excellent introductions to the many aspects of chemoinformatics that are not, as yet, of direct relevance to the processing of chemical patent information.

## 17.2 Searching Specific Chemical Structures

### 17.2.1 Representation of Chemical Structures

If one wishes to carry out computer-based searches of a chemical database then the molecules of interest must be encoded for searching, and we commence by describing the three main ways in which one can provide a full description of a chemical structure in machine-readable form: these are *systematic nomenclature*, *linear notations*, and *connection tables*. Before describing these, the reader should note that we consider here (and in the remainder of this chapter) only the processing of 2D chemical molecules, i.e., the planar chemical-structure diagrams that are conventionally used to represent molecules in the scientific literature and that are exemplified by the structure diagram shown in Fig. 17.1. More sophisticated techniques are required for the representation and searching of 3D chemical molecules, i.e., where one has geometric coordinate information for all of a molecule's constituent atoms [25].

Chemical compounds have had names associated with them ever since the days of the alchemists, but it was many years before it was realised that there was a need for systematic naming conventions to ensure that every specific molecule would have its own name. This name should be unique, in the sense that there should be only one possible name for a molecule, and unambiguous, in the sense that it should describe that molecule and no other; moreover, it was soon realised that the name should describe the various substructural components comprising the molecule, whereas common, non-systematic names will normally say little or nothing about a molecule's components. For example, 2-acetoxybenzoic acid is the systematic, explicit representation for the structure shown in Fig. 17.1, which is also, and most commonly, called aspirin.

Two systematic nomenclatures are in widespread use, these being the ones developed by the International Union of Pure and Applied Chemistry (IUPAC)<sup>1</sup> and by Chemical Abstracts Service (CAS).<sup>2</sup> IUPAC is an association of 60 national

---

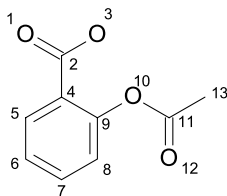
<sup>1</sup>IUPAC at <http://www.iupac.org>.

<sup>2</sup>CAS at <http://www.cas.org>.

chemical societies, seeking to establish standards in nomenclature and physiochemical data measurement, while CAS is a division of the American Chemical Society and the world's largest provider of chemical information, indexing articles from more than 10,000 journals and patents from 60 national patent agencies. Systematic names continue to be widely used in the chemical literature, but are of less importance in chemoinformatics systems, since they are normally converted automatically into one of the two other types of standard representation, i.e., linear notations or connection tables. A linear notation is a string of alphanumeric characters that provides a complete, albeit in some cases implicit, description of the molecule's topology. A *canonicalisation* procedure is normally invoked to ensure that there is a unique notation for each molecule. The first notation to be widely used was the Wiswesser Line Notation, which formed the basis for most industrial chemoinformatics systems in the 1960s and 1970s. Two notations are of importance in present-day systems: the SMILES (for Simplified Molecular Input Line Entry Specification) notation developed by Daylight Chemical Information Systems Inc. [33] and the International Chemical Identifier (or InChI), the development of which is being overseen by IUPAC. SMILES was developed for use in in-house industrial chemoinformatics systems (as is the case with much chemoinformatics software), while InChI, conversely, has been developed as an open-source, non-proprietary notation. The SMILES and the InChI for aspirin are included in Fig. 17.1.

Notations provide a compact molecular representation, and are thus widely used for compound exchange and archival purposes. However, most chemoinformatics applications will require their conversion to a connection table representation of molecular structure. A connection table is a data structure that lists the atoms within a molecule and the bonds that link those atoms together (in many cases, only heavy atoms are included since the presence of hydrogen atoms can be deduced automatically). The table provides a complete and explicit description of a molecule's topology, i.e., the way that it is connected together, whereas this information is normally only implicit in a linear notation. There are many ways in which the atoms and bonds can be encoded, with typical connection table formats being exemplified by those developed by MDL Information Systems Inc. (now Accelrys Inc.) [7]. A sample connection table for aspirin is shown in Fig. 17.1 where, for example, the first line shows that atom number 1 (Oxygen) is connected by a double bond (D) to atom number 2.

A connection table is an example of a *graph*, a mathematical construct that describes a set of objects, called *nodes* or *vertices*, and the relationships, called *edges* or *arcs*, that exist between pairs of these objects [9, 37]. This means that chemoinformatics has been able to draw on the many algorithms that have been developed previously for the processing of graphs. Of particular importance in the present context are the *graph isomorphism* algorithms that are used to determine whether two graphs are identical and the *subgraph isomorphism* algorithms that are used to determine whether one graph is contained within another, larger graph [16, 23].



InChI: 1S/C9H8O4/c1-6(10)13-8-5-3-2-4-7(8)9(11)12/h2-5H,1H3,(H,11,12)

Smiles: CC(=O)Oc1ccccc1C(=O)O

Name: 2-acetoxybenzoic acid

Connection Table:

1 O D 2
2 C D 1 S 3 S 4
3 O S 2
4 C S 2 D 5 S 9
5 C D 4 S 6
6 C S 5 D 7
7 C D 6 S 8
8 C S 7 D 9
9 C S 4 D 8 S 10
10 O S 9 S 11
11 C S 10 D 12 S 13
12 O D 11
13 C S 11

Fig. 17.1 Structure, name, InChI, SMILES and connection table for aspirin

## 17.2.2 Searching for Specific Molecules

An important search capability is structure searching: the inspection of a database to retrieve the information associated with a particular molecule (e.g., if a chemist needed to know the molecule's boiling point or to identify a synthesis for it) or to confirm the molecule's presence or absence in a database (e.g., if a chemist wanted to check whether a newly synthesised molecule was completely novel).

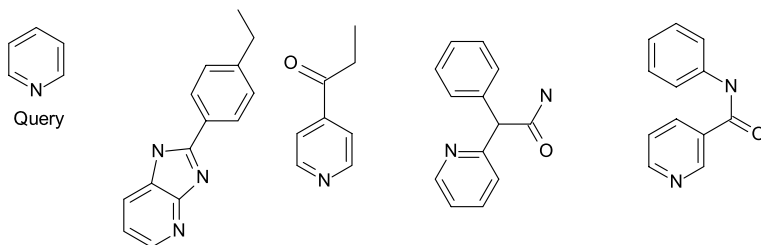
Structure searching in files of systematic nomenclature or linear notations is effected using conventional computer science techniques for single-key searching. These are typically based on hash coding, where an alphanumeric string (in this context, a systematic name or a canonicalised notation), is converted algorithmically to an integer identifier that acts as a key to the molecule's location on disk storage. A similar idea underlies the searching of connection table records; however, whereas names and notations are linear strings that can be converted into a canonical form very easily; this is not the case with connection tables and additional processing is required if hashing is to be used to enable fast structure searching. The generation of a canonical connection table requires the nodes of the chemical graph to be numbered, and there are up to  $N!$  possible sets of numberings for an  $N$ -node graph. Following initial work by Gluck [18], Morgan [26] described an algorithm to impose a unique ordering on the nodes in a graph, and hence to generate a canonical connection table that can then be used for structure searching. With subsequent development [14, 38], the resulting procedure, which is known to this day as the *Morgan algorithm*, forms the basis for all CAS databases and for many other cheminformatics systems.

Hashing is an approximate procedure, in that different records can yield the same hashed key, a phenomenon that computer scientists refer to as a *collision*. In nomenclature and notation systems, collisions are avoided by means of a subsequent, and extremely simple, string comparison that confirms the equivalence of the query molecule and the molecule that is stored in the database that is being searched. In connection table systems, a graph isomorphism algorithm is used to confirm that a true match has been achieved, this involving an exhaustive, tree-search procedure in which nodes and edges from the graph describing the query molecule are mapped to nodes and edges of the graph describing a potentially matching database molecule. The mapping is extended till all the nodes have been mapped, in which case a match has been identified; or until nodes are found that cannot be mapped, in which case the mapping backtracks to a previous, successful sub-mapping and a different mapping is attempted. A mis-match is confirmed if no match has been obtained and if there are no further mappings available for testing. It will be realised that the mapping procedure has a time complexity that is a factorial function of the numbers of graph nodes involved in the comparison, and that the procedure can thus be very demanding of computational resources. Fortunately, various heuristics are available to expedite the identification of matches, and the use of the Morgan algorithm means that very few mis-matches need to be probed, making the overall procedure rapid in operation despite the complexity of the processing that is necessary.

### 17.2.3 Searching for Chemical Substructures

Probably the single most important facility in a chemoinformatics system is the ability to carry out a substructure search, i.e., the ability to identify all of those molecules in a database that contain a user-defined query substructure. For example, in a search for molecules with antibiotic behaviour, a user might wish to retrieve all of the molecules that contain a penicillin or cephalosporin ring system. Substructure searching is effected by checking the graph describing the query substructure for inclusion in the graphs describing each of the database molecules. This is an example of subgraph isomorphism: it involves an atom-by-atom and bond-by-bond mapping procedure that is analogous to, but more complex than, that used for a graph isomorphism search. A substructure search guarantees the retrieval of all molecules matching the search criterion: unfortunately, although it is completely effective, subgraph isomorphism is extremely inefficient since it belongs to the class of NP-complete computational problems for which no efficient algorithms are known to exist [2, 23].

Operational substructure searching is practicable for three reasons. First, the fact that chemical graphs are both simple (they contain relatively few nodes, most of which are of very low connectivity) and information-rich (as one can differentiate atoms and bonds by their element and bond-types, respectively). These factors serve to reduce the numbers of atom-to-atom and bond-to-bond mappings that need to be considered by a subgraph isomorphism algorithm. Second, a lot of effort has



**Fig. 17.2** Query substructure and some example hits in a search for a pyridine ring

gone into the development of algorithms that can handle chemical graphs, as against graphs in general, very efficiently, with the elegant matching techniques described by Sussenguth [30] and by Ullmann [31] lying at the heart of current substructure searching systems. Third, and most importantly, the subgraph isomorphism search is preceded by an initial *screening search* in which each database structure is checked for the presence of features, called *screens*, that are present in the query substructure. For example, using the penicillin example mentioned above, any database structure can be eliminated from further consideration if it does not contain the fused four-membered and five-membered rings that comprise the penicillin nucleus.

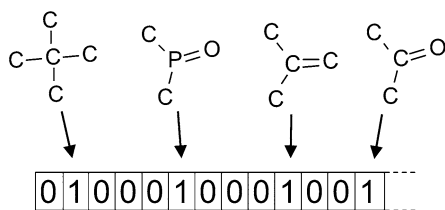
A screen is a substructural feature, called a *fragment*, the presence of which is necessary, but not sufficient, for a molecule to contain the query substructure. The features that are used as screens are typically small, atom-, bond- or ring-centred fragment substructures that are algorithmically generated from a connection table when a molecule is added to the database that is to be searched. A common example of a screen is the *augmented atom* fragment, which consists of an atom, and those atoms that are bonded directly to the chosen central atom. A representation of the molecule's structure can then be obtained by generating an augmented atom fragment centred on each atom in the molecule in turn. This information is encoded for rapid searching in a fixed-length bit-string, called a *fingerprint*, whose encoded fragments hence provide a summary representation of a molecule's structure in just the same way as a few selected keywords provide a summary representation of the full text of a document. The fingerprint representing the query can then be matched against corresponding fingerprints representing each of the molecules in the database that is to be searched. Only a very small subset of a database will normally contain all of the screens that have been assigned to a query substructure, and only this subset then needs to undergo the time-consuming subgraph isomorphism search.

Examples of substructure searching and of fingerprint generation are shown in Figs. 17.2 and 17.3, respectively.

### 17.2.4 Similarity Searching

Substructure searching provides an invaluable tool for accessing databases of chemical structures; however, it does require that the searcher is able to provide a precise definition of the substructure that is required, and this may not be possible in the

**Fig. 17.3** Example of augmented atoms and a fingerprint



early stages of a drug-discovery project, where all that is known is the identity of one or more active molecules, e.g., an existing drug from a competitor company. In such circumstances, an alternative type of searching mechanism is appropriate, called *similarity searching* [11, 35]. Here, the searcher submits an entire molecule, which is normally called the *reference structure*, and the system then ranks the database in order of decreasing similarity with the reference structure, so that the molecules returned first to the searcher are those that are most closely related to it in structural terms. The underlying rationale for similarity searching is the *Similar Property Principle* [21], which states that molecules that have similar structures will have similar properties. Hence, if the reference structure has some interesting property, such as reducing a person's susceptibility to angina, then structurally similar molecules are also likely to exhibit this characteristic.

There are many different ways, in which inter-molecular structural similarity can be quantified, with the most common similarity measures being based on the comparison of molecular fingerprints to identify the numbers of fragments common to a pair of molecules. This provides a very simple, but surprisingly, effective way of identifying structural relationships, as exemplified by the molecules shown in Fig. 17.4. However, we shall not discuss similarity searching any further here, since similarity-based approaches have not, to date, been considered in much detail for searching the generic structures that form the principal focus of this chapter. This may, of course, change in the future as techniques for searching chemical patents become more widely used and as more sophisticated searching methods become necessary for effective database access. For example, Fliri et al. [12, 13] have recently described the use of fingerprint-based similarity methods to search sets of molecules randomly enumerated from Markush structures (see Sect. 17.3.4).

## 17.3 Searching Generic Chemical Structures

### 17.3.1 Markush Structure Representation

In order to ensure complete coverage of the scope of invention, and hence protect the inventor's property rights, patent documents tend to extend beyond the realm of specific description but, instead, describe the invention using broader terms. Those features, which reflect the novelty of the invention, are described in full and unambiguous terms, whilst other features, although fundamental to the invention, may be



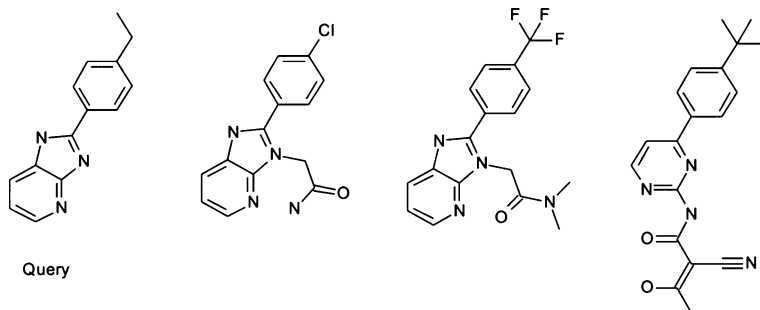


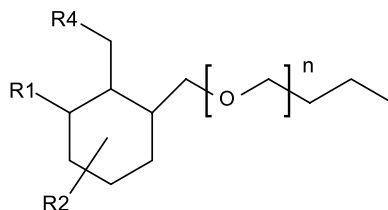
Fig. 17.4 Example of output from a similarity search

optional or alternative in nature. An example of the latter feature might be a new refrigerator for which the internal light might be described using a vague term such as “device for illuminating the interior”. The same is true of chemical patents in which features of the compound, that are fundamental to its novelty of operation are described using specific terms, and those for which alternatives may be substituted are described generically. The result of this treatment is a single description, which can represent a potentially vast number of specific molecules, many (or even most) of which will have never been synthesised or tested.

The logical and linguistic terminology that exists in the chemical patent literature has been described in detail by Dethlefsen et al. [8], leading to a classification of the structural variations that exist. These authors identified four types of structural variation, which are exemplified in Fig. 17.5. Substituent variation involves the (possibly optional) set of alternative components, which may be attached at a fixed point of substitution (e.g., R1 in the figure); position variation involves the alternative positions of attachment between two components of the molecule (e.g., R2). Frequency variation involves the repetition of a component either within a linear sequence or as an attachment to a ring system (e.g.,  $n$ , indicating the presence of between one and two occurrences of the  $-O-CH_2-$  substructure); and homology variation involves the use of terminology which is itself generic in nature and which defines the component as being a member of a family of related chemical substituents (e.g., R4 in the figure indicating an alkyl group member containing one, two or three carbon atoms).

Figure 17.5 illustrates a relatively simple generic structure, but repeated nesting of alternative components within parent components is a common feature in chemical patents, leading to a complex and often confusing structure. Enumeration of all of these the specific molecules is rarely an option due to storage requirements and computational costs. Therefore, an alternative method of computer representation is required. The basic structure adopted by current commercial systems [5] is a logical tree in which the invariant core of the structure, the graphical component in Fig. 17.5 for example, becomes the root. The various optional and alternative components become the branches of the tree, and the logical and connective relationships are maintained within the representation [4], as exemplified in Fig. 17.6.

**Fig. 17.5** Examples of structure variation in generic chemical structures



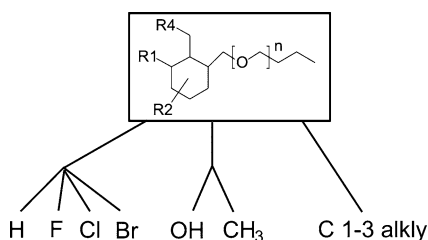
R1 is optionally F, Cl or Br

R2 is OH or CH<sub>3</sub>

R4 is C 1-3 alkyl

n = 1-2

**Fig. 17.6** Tree representation of a generic structure



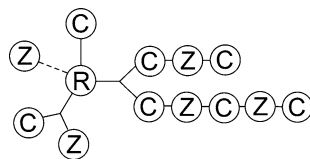
The logical tree encodes all of the linkages, potential or actual, within the set of molecules covered by a Markush structure, and it can hence be regarded as a form of connection table, albeit one that is far more complex than that used to describe a single specific molecule.

### 17.3.2 Representational Transparency

The representation of the components themselves in the tree depends on whether they are specific or generic in nature, the latter being an instance of homology variation. Specific components can be represented by a connection table, or even a line notation, whereas components relating to a chemical family, or homologous series, require alternative means. In the latter case, the representation is usually a single node which may be labelled according to the family group, and which is usually qualified by further attributes such as the number of carbon atoms or number of rings present. In the Markush DARC system, which originated from a collaboration between Derwent Publications and the French Patent Office INPI, (now called the Merged Markush Service, MMS, and produced by Thomson Reuters) these are termed “Superatoms”, whilst the MARPAT system produced by CAS uses “Hierarchical Generic Groups”.

Whichever method is employed, there remains the problem of *transparency* between the two types of representation, i.e., the lack of a common representation across components. During a search operation, whether for a structure or for a sub-structure, the aim is to identify mappings between the components of the query

**Fig. 17.7** Reduced graph representation of the generic structure of Fig. 17.6 (optional connections are indicated by a dotted line)



structure and those of the database structure. This operation is complicated by the requirement to map features which are specific in one representation with those which may be generic in others, a one-to-many mapping, or even features which are generic in both. In order to overcome this transparency problem, a common representation is usually sought so that the mapping becomes like-for-like. The enumeration of all possible specific members of the homologous series is again usually not an option, so a more appropriate step is the aggregation of specific components into their respective generic nodes. In the Sheffield Generic Structures Project [24], several aggregation methods were investigated, leading to a transparent representation called a *reduced graph* [17]. Figure 17.7 illustrates an example of such a graph in which aggregation is based on the ring (R) or non-ring nature of the features, and on further subdividing the non-ring features into those that are all carbon (C) and those that are non-carbon (Z).

Since we now have a common representation, one-to-one mapping can be carried out between the query and database structure. The final, and now less complex, stage is to map the constituent features of the matching query node and the database node. These are still likely to contain generic and/or specific components, but the operation is now more localised and much simpler and can be implemented using a modified version of Ullmann's subgraph isomorphism algorithm [19, 31].

### 17.3.3 Fragmentation Codes and Screening

Early structure-based retrieval systems operated almost exclusively on the basis of fragmentation codes in which the structural components were described using a series of fragment descriptors that were analogous in principle to the fragments used for screening substructure searches of databases of specific molecules. The most notable fragmentation codes were the Derwent Chemical Code used by Derwent Publications Ltd. [28], the DuPont/IFI code [22] and the GREMAS code from International Documentation in Chemistry [29]. The GREMAS system was highly effective and it was later possible to generate the codes automatically from the structure representation [27].

As with specific structure searching, graph-based generic systems, such as MARPAT and Markush DARC, also require an initial fragment-based screening stage in order to reduce the number of compounds being sent to more computer intensive search strategies. In addition to the standard screens used at CAS for searching specific molecules, the MARPAT system uses generic group screens in which the components are reduced to their Hierarchical Generic Groups. The Markush

DARC system also extended their existing specific search screens with the addition of Fuzzy FRELs (where a FREL is a circular fragment that can be considered as a larger version of the augmented atom discussed previously); some of these fuzzy FRELs were defined in terms of Superatoms and others reflected specific local variations. In the system developed at Sheffield, the approach was to generate specific fragment descriptors from the generic components [20]. Two types of screen were developed: those from the invariant components of the molecule, i.e. those alternatives which are common to all molecules covered by the generic; and those which would be optional depending on the individual specific molecules being considered at any point. In Fig. 17.5, for instance, a screen denoting a halogen would be common to all molecules, with a logical “bubble-up” of all screens from the branches of the tree to its root maintaining the logical relationships between screens [10].

### 17.3.4 Recent Developments

More recently, there has been renewed interest in Markush structures; in part due to increased computer power, which was not available when the current systems first evolved. One area of interest is the application of Oracle relational database systems for storing and searching Markush structures [3, 6]. Many of the new developments do not, however, deal with all types of structure variation, and rely on the same philosophy of extending current systems for handling specific chemical structures.

Two other areas of interest are the automatic extraction of structural information from the patent documents [32, 36, 39] and enumeration of specific compounds from the Markush structure. Chemical patent documents contain structures for the specific claim as well as a selection of examples. Although these usually represent a very small proportion of the possibly infinite number of compounds represented by the Markush structure, they are clearly a rich source of information and are indexed accordingly. A further source of structural information comes from the translation of nomenclatural terms identified in the document, as in the SureChem database and search system.<sup>3</sup> Full enumeration of all represented compounds is not possible for most structures due to the combinatorial complexity. However, as noted previously, sets of randomly enumerated specifics have been used for similarity searching, enabling rapid patent analysis and virtual library creation [12, 13].

## 17.4 Conclusions

The structures of chemical molecules are an important component of the information contained in chemical patents. Individual molecules can be searched using

---

<sup>3</sup><http://www.surechem.org>.

well-established techniques from chemoinformatics, and substantial enhancements to these techniques have allowed them to be used for the representation and searching of the generic chemical structures in patents, which can describe very large numbers of structurally related molecules. In this chapter, we have summarised the techniques that are currently available for structure and substructure searching of both specific and generic structures. There are, however, many problems that remain to be addressed. Most importantly, the very generic descriptions that are sometimes used in patents mean that very large hit-lists can result even in response to quite specific structural queries: it is hence likely that there will be much interest in the future in the use of similarity-based procedures to rank search-outputs so that attention can be focussed on just the top-ranked structures and patents.

## References

1. Barnard JM (ed) (1984) Computer handling of generic chemical structures. Gower, Aldershot
2. Barnard JM (1993) Substructure searching methods—old and new. *J Chem Inf Comput Sci* 33:532–538
3. Barnard JM, Wright PM (2009) Towards in-house searching of Markush structures from patents. *World Pat Inf* 31:97–103
4. Barnard JM, Lynch MF et al (1982) Computer storage and retrieval of generic structures in chemical patents. Part 4. An extended connection table representation for generic structures. *J Chem Inf Comput Sci* 22:160–164
5. Berks AH (2001) Current state of the art of Markush topological search systems. *World Pat Inf* 23:5–13
6. Csepregi S, Mate N, Csizmadia F et al (2009) Representation, searching & enumeration of Markush structures—from molecules towards patents. <http://www.chemaxon.com/library/scientific-presentations/calculator-plugins/representation-searching-enumeration-of-markush-structures-from-molecules-towards-patents-2009-update/>. Accessed 14 September 2010
7. Dalby A, Nourse JG et al (1992) Description of several chemical structure file formats used by computer programs developed at Molecular Design Limited. *J Chem Inf Comput Sci* 22:244–255
8. Dethlefsen W, Lynch MF et al (1991) Computer storage and retrieval of generic chemical structures in patents, Part 11. Theoretical aspects of the use of structure languages in a retrieval system. *J Chem Inf Comput Sci* 31:233–253
9. Diestel R (2000) Graph theory. Springer, New York
10. Downs GM, Gillet VJ et al (1989) Computer storage and retrieval of generic chemical structures in patents, Part 10. Assignment and logical bubble-up of ring screens for structurally explicit generics. *J Chem Inf Comput Sci* 29:215–224
11. Eckert H, Bajorath J (2007) Molecular similarity analysis in virtual screening: foundations, limitation and novel approaches. *Drug Discov Today* 12:225–233
12. Fliri A, Moysan E et al. (2009) Methods for processing generic chemical structure representations. US Patent 2009/0132464
13. Fliri A, Moysan E, Nolte M (2010) Method for creating virtual compound libraries within Markush structure patent claims. WO Patent 2010/065144 A2
14. Freeland R, Funk S et al (1979) The chemical abstracts service chemical registry system. II. Augmented connectivity molecular formula. *J Chem Inf Comput Sci* 19:94–98
15. Gasteiger J (2006) The central role of chemoinformatics. *Chemom Intell Lab Syst* 82:200–209
16. Gasteiger J, Engel T (eds) (2003) Chemoinformatics: A textbook. Wiley-VCH, Weinheim

17. Gillet VJ, Downs GM et al (1987) Computer-storage and retrieval of generic chemical structures in patents. 8. Reduced chemical graphs and their applications in generic chemical-structure retrieval. *J Chem Inf Comput Sci* 27:126–137
18. Gluck DJ (1965) A chemical structure storage and search system developed at DuPont. *J Chem Doc* 5:43–51
19. Holliday JD, Lynch MF (1995) Computer storage and retrieval of generic chemical structures in patents. Part 16. The refined search: an algorithm for matching components of generic chemical structures at the atom-bond level. *J Chem Inf Comput Sci* 35:1–7
20. Holliday JD, Downs GM et al (1993) Computer storage and retrieval of generic chemical structures in patents, Part 15. Generation of topological fragment descriptors from nontopological representation of generic structure components. *J Chem Inf Comput Sci* 33:369–377
21. Johnson MA, Maggiora GM (eds) (1990) Concepts and applications of molecular similarity. Wiley, New York
22. Kaback SM (1984) The IFI/Plenum chemical indexing system. In: Barnard JM (ed) Computer handling of generic chemical structures. Gower, Aldershot
23. Leach AR, Gillet VJ (2007) An introduction to chemoinformatics. Kluwer, Dordrecht
24. Lynch MF, Holliday JD (1996) The Sheffield generic structures project—a retrospective review. *J Chem Inf Comput Sci* 36:930–936
25. Martin YC, Willett P (eds) (1998) Designing bioactive molecules: Three-dimensional techniques and applications. American Chemical Society, Washington
26. Morgan H (1965) The generation of a unique machine description for chemical structures—a technique developed at Chemical Abstracts Service. *J Chem Doc* 5:107–113
27. Rössler S, Kolb A (1970) The GREMAS system, an integral part of the IDC system for chemical documentation. *J Chem Doc* 10:128–134
28. Simmons ES (1984) Central patents index chemical code: a user's viewpoint. *J Chem Inf Comput Sci* 24:10–15
29. Suhr C, von Harsdorf E, Dethlefsen W, Derwent's CPI (1984) IDC's GREMAS: remarks on their relative retrieval power with regard to Markush structures. In: Barnard JM (ed) Computer handling of generic chemical structures. Gower, Aldershot
30. Sussenguth EH (1965) A graph-theoretic algorithm for matching chemical structures. *J Chem Doc* 5:36–43
31. Ullmann JR (1976) An algorithm for subgraph isomorphism. *J ACM* 23:31–42
32. Valko AT, Johnson AP (2009) CLiDE Pro: The latest generation of CLiDE, a tool for optical chemical structure recognition. *J Chem Inf Comput Sci* 49:780–787
33. Weininger D (1988) SMILES, a chemical language and information-system. 1. Introduction to methodology and encoding rules. *J Chem Inf Comput Sci* 28:31–36
34. Willett P (2008) From chemical documentation to chemoinformatics: fifty years of chemical information science. *J Inf Sci* 34:477–499
35. Willett P (2009) Similarity methods in chemoinformatics. *Ann Rev Inf Sci Technol* 43:3–71
36. Williams AJ, Yerin A (2009) Automated identification and conversion of chemical names to structure-searchable information. In: Banville DL (ed) Chemical information mining. CRC Press, Boca Raton
37. Wilson R (1996) Introduction to graph theory. Longman, Harlow
38. Wipke WT, Dyott TM (1974) Stereochemically unique naming algorithm. *J Am Chem Soc* 96:4825–4834
39. Zimmermann M, Thi LTB, Hofmann M (2005) Combating illiteracy in chemistry: towards computer-based chemical structure reconstruction. *ERCIM News* 60:40–41

# Chapter 18

## Offering New Insights by Harmonizing Patents, Taxonomies and Linked Data

Andreas Pesenhofer, Helmut Berger, and Michael Dittenbach

**Abstract** Patent classification schemes such as the International Patent Classification maintained by the World Intellectual Property Organization are of vital importance for patent searchers, because they usually act as an entry point for the search process. We present methods for augmenting patents by assigning them to classes of a different classification scheme, i.e. a science taxonomy derived from the Wikipedia Science Portal. For each scientific discipline contained in the portal, descriptive keywords are extracted from the linked Web pages. These keywords are used to identify relevant patents and associate them to the appropriate scientific disciplines. Additional to that we augment the patents with data sets from the Linking Open Data (LOD) project. The ontology and the data sets of the LOD cloud are part of the Patent Taxonomy Integration and Interaction Framework, which is a flexible approach allowing for the integration of different patent ontologies enabling a wide range of interaction methods.

### 18.1 Introduction

Finding all patents relevant to a particular invention in the vast amount of documents available in the many existing patent databases is a difficult task. Moreover, missing just a single relevant patent, and thus violating intellectual property (IP) rights of others, can be very expensive for a company when introducing a new product on the market that uses an already patented technology. Thus, professional patent searchers are forced to read (or at least skim through) all retrieved documents, because the relevant one could be at the bottom of the search result list. This clearly contrasts Web users posing ad-hoc queries to Web search engines who hardly ever look further than at the ten top-ranked results. Patent classification systems such as

---

A. Pesenhofer (✉) · H. Berger · M. Dittenbach  
max.recall information systems, Vienna, Austria  
e-mail: [a.pesenhofer@max-recall.com](mailto:a.pesenhofer@max-recall.com)

H. Berger  
e-mail: [h.berger@max-recall.com](mailto:h.berger@max-recall.com)

M. Dittenbach  
e-mail: [m.dittenbach@max-recall.com](mailto:m.dittenbach@max-recall.com)

the International Patent Classification (IPC) maintained by the World Intellectual Property (WIPO) are important structural instruments for organizing patents into taxonomies of technology domains. These taxonomies allow searchers to constrain search queries to particular technological fields in order to reduce the amount of documents to be read.

However, these classification schemes have been built by experts for experts. A more general taxonomy would make it easier for non-expert users of patent information systems to access and navigate through the information space. Such non-experts might not be familiar with the definitions of the section, classes and subclasses, groups and subgroups contained in the IPC, but usually they are familiar with the subject matter of the invention or an idea itself, i.e. the scientific background. To this end, we have created such a taxonomy, the Wikipedia Science Ontology (WikiSCION), based on the Wikipedia Science Portal, which is a well-maintained starting point for a top-down discovery of *Science* as topic of interest and its many subtopics. Note that a taxonomy is a restricted ontology with a well defined hierarchy and a single root node. The Wikipedia can be seen as a source of knowledge crafted, shaped and maintained by a large community agreeing on a language that is supposed to be easier to comprehend by non-experts. We have developed a method for automatically assigning patent documents to the classes of this taxonomy, and thus enriching the patents with additional meta-information.

In particular, we use the information contained in the Wikipedia pages dedicated to the various scientific disciplines to extract relevant keywords. These keywords are then used to associate patents relevant to the scientific disciplines. The Linked Data principals are used for enriching the patents with the links to the Wikipedia Science Ontology. We have selected data sets being part of the Linked Open Data (LOD)<sup>1</sup> cloud for adding additional metadata to the patents. The WikiSCION is part of the Patent Taxonomy Integration and Interaction Framework (PTI2), which is a flexible approach allowing for the integration of different ontologies and data sets of the LOD cloud enabling a wide range of interaction methods.

The remainder of this chapter, being an extended version of [5], is structured as follows. In Sect. 18.2, we outline selected related work. Section 18.3 describes the PTI2 framework with special focus on the Wikipedia Science Ontology and the integration of LOD. Then, we exemplify our method by describing the Web-based user interface in Sect. 18.4 followed by several conclusions in Sect. 18.5.

## 18.2 Related Work

The International Patent Classification is a standard taxonomy developed and administered by the World Intellectual Property Organization for classifying patents

---

<sup>1</sup><http://linkeddata.org/>.



and patent applications. IPC describes a wide range of topics covering human inventions and relies on a diverse technical and scientific vocabulary. A large part of IPC is concerned with chemistry, mechanics, and electronics. Thus, the IPC is a complex, hierarchical taxonomy that has been maintained and refined for more than 30 years (cf. [3]).

Yu et al. [11] performed a task-based ontology evaluation using existing measures proposed in literature for a real world application—the Wikipedia and its categories. In their studies they model the task of the browsing of an information space using a given category structure that originates from the English-language version of Wikipedia and its associated articles. They found out that tangledness, which helps to understand how intersected the category structure is, may be desirable in ontologies and category structures for browsing in general knowledge application areas like Wikipedia. This was especially significant in tasks that required specific information. They also studied a method for generating a category structure but generally found that it was not comparable to the Wikipedia version.

Sah and Hall [7] developed a semantic portal, referred to as SEMPort. It provides personalized views, semantic navigation, ontology-based search and three different kinds of semantic hyperlinks. The portal offers a Web interface for distributed content editing and provision of ontologies in real-time. The system was tested on the Course Module Web Page of the School of Electronics and Computer Science in Southampton where the different browsing needs of the users (students and teachers) were modeled. The first outcome was that, the system alleviates problems associated with navigation through personalized views, semantic hyperlinks, semantic navigation and ontology-based search, detailed user studies have not been carried out yet.

The goal of PATExpert [9], an advanced patent processing service, is to push forward the adoption of the semantic paradigm for patent processing and to provide a user technique allowing for more powerful access to the content of textual patent documents. They introduce a content representation scheme for patent documentation and sketch the design of techniques that facilitate the integration of this scheme into the patent processing cycle. Two types of techniques are discussed. Techniques of the first type facilitate the access to the content of patent documentation provided in a textual format—be it by the human reader or by the machine—in that they rephrase and summarize the documentation and map it onto a formal semantic representation. Techniques of the second type operate on the content representation.

An evaluation of the cooperation and quality in Wikipedia was carried out by Wilkinson and Huberman [10]. They have shown that high-quality articles in Wikipedia are distinguished from the rest by a larger number of edits and distinct editors, by their article visibility, popularity, and age. Furthermore, they demonstrated more intense patterns of cooperation in the high-quality articles than in other articles. These findings are in contrast to observations of cooperative efforts in other domains where result quality does not necessarily increase with the number of collaborators. The article growth follows a very simple overall pattern on average. This pattern implies that a small number of articles, corresponding to topics of high relevance, accrete a disproportionately large number of edits, while the vast majority of articles show far less activity.

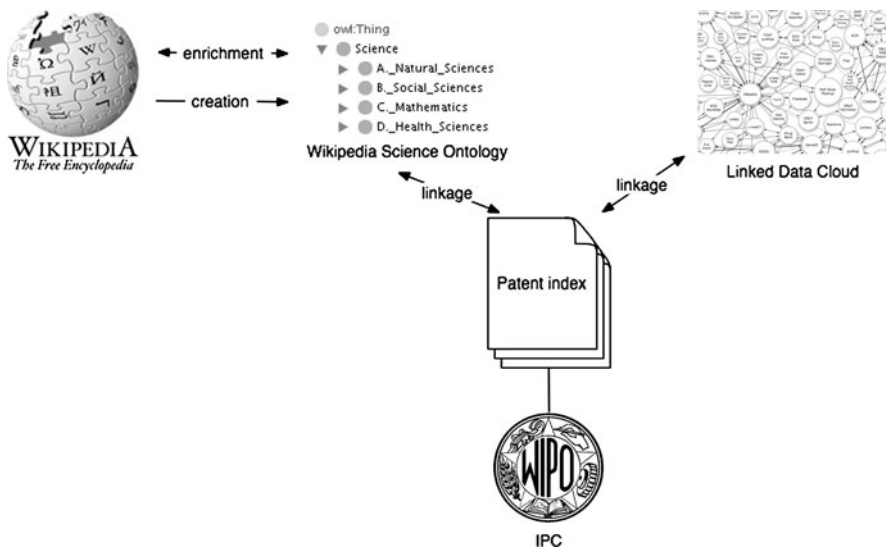
Zirn et al. [12] evaluated an automatic method for differentiating between instances and classes in a large-scale taxonomy induced from the Wikipedia category network. The method exploits characteristics of the category names and the structure of the network. Their approach is a first attempt to perform an automated distinction in a large-scale resource.

During the last years the Semantic Web which provides a common framework allowing data to be shared and reused across application, enterprise, and community boundaries has gained on interest. The basic idea of linked data was outlined by Tim Berners-Lee [1], who defined four rules for the web of data constructed with documents on the web. The goal of the W3C SWEO Linking Open Data community project is to extend the Web with a data commons by publishing various open data sets as RDF on the Web and by setting RDF links between data items from different data sources [13]. The Resource Description Framework (RDF) allows to describe resources (in particular Web resources) in the form of subject-predicate-object expressions, which is the base data structure in the Semantic Web. In November 2009, the LOD project counted more than 13.1 billion RDF triples, where a triple is a piece of information that consists of a subject, predicate, and object to express a particular subject's property or relationship to another subject. In addition the LOD project interlinked around 142 million RDF links. These links enable the user to navigate from a data item within one data source to related data items within other sources using a Semantic Web browser. RDF links can also be followed by the crawlers of Semantic Web search engines, which may provide sophisticated search and query capabilities over crawled data. The query results are structured data and not just links to HTML pages, which can be used within other applications. The number of open data sets is rapidly growing, at time of writing e.g. Wikipedia, Wikibooks, Geonames, MusicBrainz, WordNet, the DBLP bibliography and the US government portal data.gov are among many other data sets available.

A survey of current research efforts in representing biomedical knowledge in Semantic Web languages is given by Sougata Mukherjea [4]. He describes in detail the Semantic Web languages and discusses current efforts to represent biomedical knowledge in these languages. Here the Gene Ontology and the Unified Medical Language System (UMLS) served as source of information. As a possible application that uses the biomedical Semantic Web he gives the example of using patents, their inventors and assignees as well as all UMLS biomedical concepts as resources. This allows him to formulate a query that finds all inventor and assignee pairs who have a patent which has a term belonging to one specific UMLS class. He concludes that it is a challenge to develop a biomedical Semantic Web that stores most, if not all, of the biomedical knowledge and that another major problem is scalability.

### 18.3 Patent Taxonomy Integration and Interaction Framework

The conceptual design of the Patent Taxonomy Integration and Interaction Framework is depicted in Fig. 18.1. In this approach, the Wikipedia, more precisely the



**Fig. 18.1** Conceptual design of the patent taxonomy integration and interaction framework

```

<!ELEMENT class ( name | own_slot_value | superclass |
template_facet_value | template_slot | type )* >
<!ELEMENT facet_reference ( #PCDATA ) >
<!ELEMENT knowledge_base ( class+, slot+, simple_instance+ ) >
<!ATTLIST knowledge_base xsi:schemaLocation CDATA #REQUIRED >
<!ELEMENT name ( #PCDATA ) >
<!ELEMENT own_slot_value ( slot_reference, value+ ) >
<!ELEMENT simple_instance ( name, type, own_slot_value+ ) >
<!ELEMENT slot ( name, type+, own_slot_value ) >
<!ELEMENT slot_reference ( #PCDATA ) >
<!ELEMENT superclass ( #PCDATA ) >
<!ELEMENT template_facet_value ( slot_reference, facet_reference,
value+ ) >
<!ELEMENT template_slot ( #PCDATA ) >
<!ELEMENT type ( #PCDATA ) >
<!ELEMENT value ( #PCDATA ) >
<!ATTLIST value value_type ( class | string ) #REQUIRED >
    
```

**Fig. 18.2** Scheme of WikiSCION

science disciplines, are the starting point for deriving the Wikipedia Science Ontology. WikiSCION is a structured representation of 1,113 science disciplines, which has been compiled by a human expert. The scheme underlying the ontology is inspired by the ACM Computer Classification System and is depicted in Fig. 18.2. In the `template_slot` our specific properties `has_WikiURL`, `has_Terms`, `has_Acronym`, `has_Synonym`, `label`, `seeAlso` and `comment` were modeled.

After the science relationships have been captured in the ontology, an automatic enrichment process for adding significant keywords describing the science disciplines has been carried out. To this end, those keywords were extracted from the respective Wikipedia page, which were not present on any other page and associated with the corresponding discipline. On average, 31 keywords were associated to each discipline. Additionally, a large set of US Patent and Trademark Office (USPTO) patent documents has been preprocessed and an index has been generated. This index was used to align the patent documents with the science disciplines. More specifically, the similarity between the keywords describing the science discipline and the words found in the patents was calculated. A threshold value was used to identify the most relevant documents, i.e. if the similarity score is above a certain value, it is regarded as being relevant, otherwise it is ignored.

The Linked Data Cloud was used as a second data source. We focused on data of DBpedia,<sup>2</sup> a community effort to extract structured information from Wikipedia and to make this information available on the Web, and linked this data to our patent corpus. Finally, the PTI2 framework offers a Web interface for browsing the Wikipedia Science Ontology and a RDF representation of the patents with links to the Wikipedia Science Ontology and links to resources of the LOD cloud.

The following sections give a detailed description of the elements and processing steps needed in order to develop a prototype of PTI2.

### 18.3.1 *The Wikipedia Science Ontology*

The Wiki concept, invented by Ward Cunningham in 1995, is based on the principles of freely added information and associations by any user. Content generation in Wikis follows the concepts of a more or less global brainstorming in computer science known as “Distributed Asynchronous Collaboration” (cf. [2]). Precup et al. [6] describe distributed asynchronous collaboration as collaboration between geographically distributed people, offering asynchronous access to explicit knowledge and information that is produced by means of technologies such as fax, phone or e-mail. The conceptual formulation creating a human-centered, intuitive navigation structure of science for a patent ontology was the reason for choosing a collaboratively grown, large and free text corpus, the Wikipedia, which covers all encyclopedic topics and is the result of global human brainstorming.

We decided to start with the definitions found in the Wikipedia Science Portal, since Wikipedia Portals show more editorial accuracy and critical engagement than other parts of Wikipedia. At the time of writing, Wikipedia offers 548 Portals, whereof 119 are *featured portals*<sup>3</sup> with general quality standards.

---

<sup>2</sup><http://dbpedia.org/>.

<sup>3</sup>[http://en.wikipedia.org/wiki/Wikipedia:What\\_is\\_a\\_featured\\_portal](http://en.wikipedia.org/wiki/Wikipedia:What_is_a_featured_portal).

The Wikipedia Science Ontology is the backbone of our system. It is intended to represent all science disciplines found in the Wikipedia Science Portal. The generation was done by reading continuous text in order to build a mono-hierarchic taxonomy. Note that in the context of collaboratively edited content in Wikipedia, issues such as uncontrolled terminology, “fluid” content (permanently changing versions), inconsistency in content, reliability of content and authors became evident.

The following procedure was carried out in order to create the initial version of the Wikipedia Science Ontology.

- Identification of the top-level concept *Science*
- Definition of a starting point for reading: To ensure the traceability of this task, the main page of the Wikipedia Science Portal<sup>4</sup> was selected as the starting point.
- Capturing the first mono-hierarchic structure of WikiSCION: The initial mono-hierarchic taxonomy is generated by association distinct numbers with each science discipline. As an example consider *Microbiology*<sup>5</sup> and its sub-disciplines. They were identified by manually extracting them from the content of the page. Eleven sub-disciplines were identified:

2.1.2.3 Microbiology

2.1.2.3.1 Microbial physiology

2.1.2.3.2 Microbial genetics

2.1.2.3.3 Medical microbiology

2.1.2.3.4 Veterinary microbiology

2.1.2.3.5 Environmental microbiology

2.1.2.3.6 Evolutionary microbiology

2.1.2.3.7 Industrial microbiology

2.1.2.3.8 Aeromicrobiology

2.1.2.3.9 Food microbiology

2.1.2.3.10 Pharmaceutical microbiology

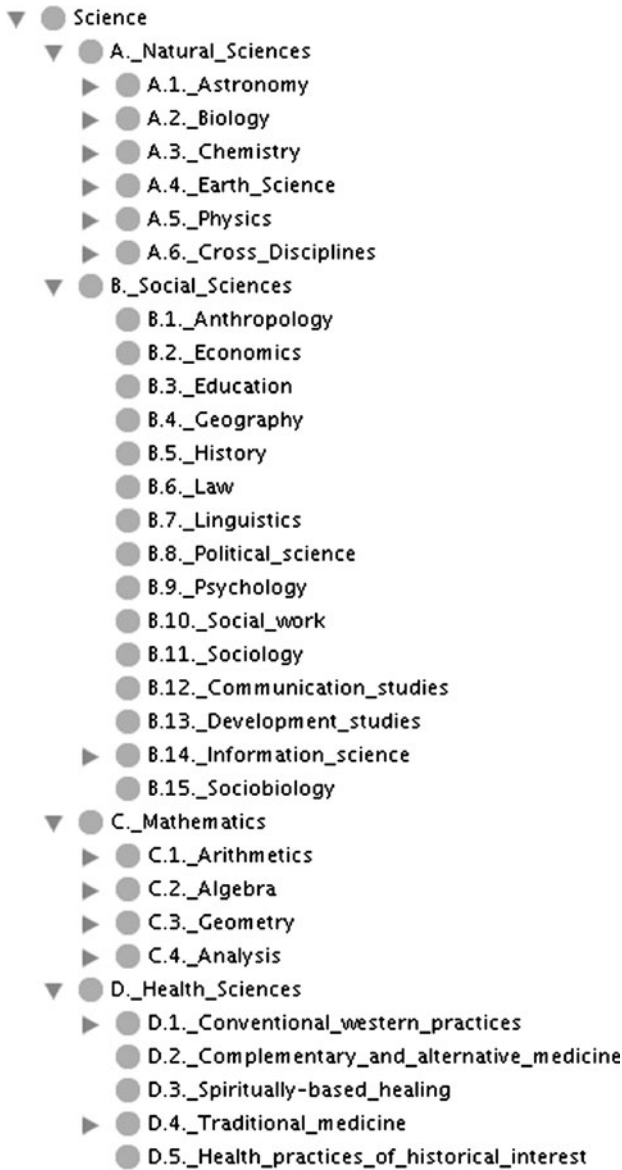
2.1.2.3.11 Oral microbiology

- Consistency analysis and refinement: When developing knowledge bases containing controlled vocabulary and semantic relationships it is obligatory to follow a consistent terminology defined by a strict rule set. Such knowledge bases are generated and maintained over decades in joint efforts of thousand of librarians following standardized rules for term definition and building relationships. The Anglo-American Standard is the Library of Congress Subject Headings (LCSH), which is an international standard reference database. The German counterpart is the *Schlagwortnormdatei* (SWK). These knowledge bases consist of norm terms and their relations defined by a human indexer. However, building a standard out of a heterogeneous text corpus is a complicated task. One of the reasons for this are contradictory statements found on Wikipedia pages [8].

---

<sup>4</sup><http://en.wikipedia.org/wiki/Portal:Science>.

<sup>5</sup><http://en.wikipedia.org/wiki/Microbiology>.



**Fig. 18.3** First and second level of the science ontology in Protégé

The approach for systematic compilation of scientific disciplines as outlined above was applied on large scale to the Wikipedia Science Portal. The ontology was created using Protege and stored in OWL representation (see Fig. 18.3).

### 18.3.2 *Ontology Enrichment*

This section describes the automatic enrichment of the manually created ontology by (1) deriving a Wikipedia page from the `label` property, (2) checking if the page exist and finally (3) computing the specific terms for all leaf nodes of the ontology.

#### 18.3.2.1 Derivation of the Link

In the creation phased of the ontology for each class a property named `label` was added. This property corresponds to the title of the Wikipedia page. For some classes the property `has_WikiURL` was entered during the manual creation of the ontology. This property specifies the Uniform Resource Locator (URL) of the corresponding Wikipedia page. For those classes where no WikiURL existed, it was automatically generated using a predefined rule set. For example, the string `Spina_bifida` was created for the label value *spina bifida*, and was then appended to the Wikipedia base URL (<http://en.wikipedia.org/wiki/>), resulting in [http://en.wikipedia.org/wiki/Spina\\_bifida](http://en.wikipedia.org/wiki/Spina_bifida).

#### 18.3.2.2 Checking the Existence of the Page

In the next step, it was verified whether the created Wikipedia URL exists (cf. Fig. 18.4) or not. If a page did not exist, the property `has_WikiURL` was removed. This was the case for 121 of 915 pages.

#### 18.3.2.3 Extraction of Descriptive Terms

In the current implementation we limit the calculation of descriptive terms to the leaf nodes of the ontology, i.e. nodes without subclasses. For content indexing, only the text between the HTML division tag with the id `content` was used. The pages were downloaded and indexed separately using Lucene,<sup>6</sup> i.e. one index per leaf node of the ontology was generated. Then, a term list for each leaf index was generated. For a particular ontology class, only those terms were taken into account that, first, consist only of lower-case characters, and second, only occur in a single leaf index. The terms were stored in the `has_Terms` property of the this class.

An example of extracted terms for the class *Spina bifida* is given in [Appendix](#). Please note that the terms *spina* and *bifida* are not included in the list when a document frequency of 1 is used.

---

<sup>6</sup><http://lucene.apache.org/>.



Fig. 18.4 Wikipedia page for Spina bifida

### 18.3.3 Patent Linking

Our corpus consists of 1,330,841 patent applications published by the USPTO between 2001 and 2006. The documents are stored in XML format and were indexed using Lucene allowing for keyword-based searching. We used the standard tokenization method, which converts all characters to lower case and removes stop words.

For each leaf node of the WikiSCION, a query string was created using the terms of the `has_Terms` property connected by the Boolean operator `OR`. With this query, the patent index was searched. The returned documents were ranked by relevance and associated with the leaf node of the ontology by storing the information in the `has_Patents` property.

Parts of the patents are now available as a separate resource described in RDF. This RDF description is enriched with the links to the WikiSCION ontology.

### 18.3.4 Patent Enrichment with Linked Data

The patents do not contain any annotations that follow the principals of the Semantic Web. Therefore, we downloaded the geographic coordinates and the raw infobox properties data set of DBpedia. The unique subjects were extracted and also tokenized in order to separate all terms by a blank. This list was then used to formulate the queries that were run against the Lucene index of the patent corpus. In case of a match, the patent and the corresponding data resource were liked by adding the RDF triples to the resource description of the patent. This enrichment process was carried out for both data sets from DBpedia.



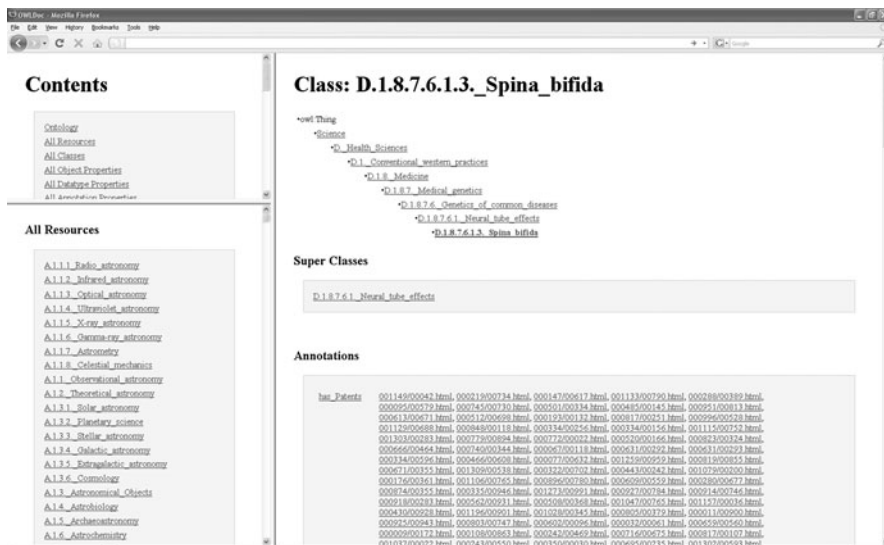


Fig. 18.5 User interface for navigating throughout the WikiSCION

## 18.4 User Interface

We have developed a first Web-based user interface for browsing the WikiSCION. This interface is split into three parts: (i) a high-level view on the ontology in the top-left corner, (ii) the taxonomy of sciences in the lower-left corner, and (iii) the associated patent on the right-hand side. As an example the view on the concept *Spina bifida* is depicted in Fig. 18.5.

Wikipedia gives the following definition for *Spina bifida*<sup>7</sup>:

*Spina bifida* (Latin: “split spine”) is a developmental birth defect involving the neural tube: incomplete closure of the embryonic neural tube results in an incompletely formed spinal cord. [...] Most affected individuals will require braces, crutches, walkers or wheelchairs to maximize their mobility. The higher the level of the *spina bifida* defect the more severe the paralysis. [...]

When navigating to the class *Spina bifida*, which is a sub-class of *Medical Genetics*, the following patent is part of the result set: “Pedaling aid for handicapped musician”. This patent describes a pedaling aid combined with an acoustic piano, and assists a physically handicapped person in performing a piece of music on the acoustic piano (Patent no.: US20060112809). Note that the term *Spina bifida* does not occur in the document, but the user is pointed to the relevant scientific context. Interestingly, this particular patent is solely associated with the class *Spina bifida* in the Science Ontology.

The second patent entitled “Tray supporting device for a wheelchair” also describes an invention that is relevant for handicapped people. We want to note that a

<sup>7</sup>[http://en.wikipedia.org/wiki/Spina\\_bifida](http://en.wikipedia.org/wiki/Spina_bifida).

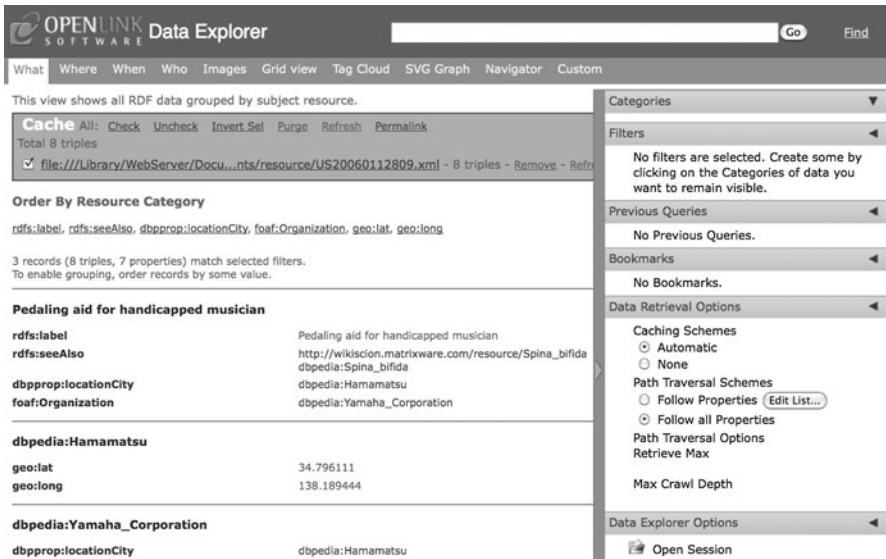


Fig. 18.6 RDF data for an enriched patent

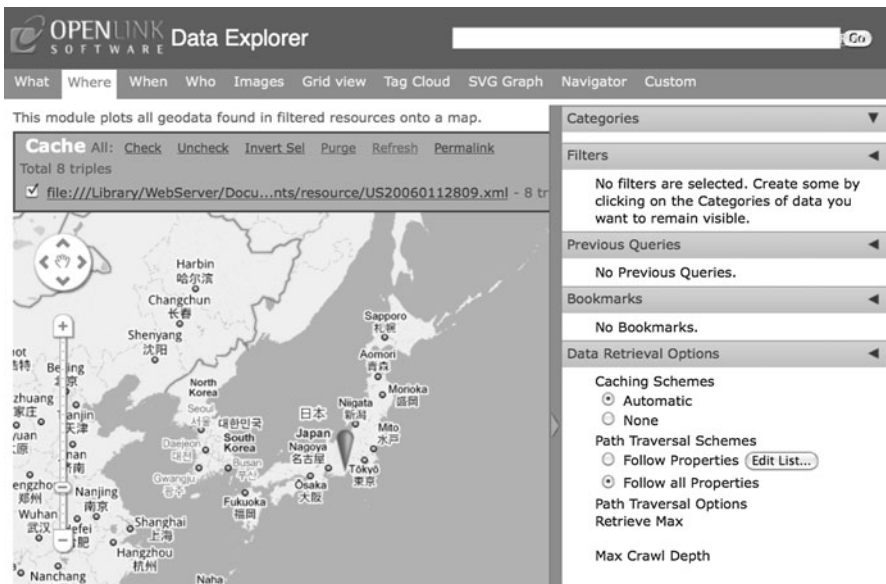


Fig. 18.7 Origin of the patent

first manual inspection showed promising results, however, we plan to conduct an in-depth evaluation in the future.

<b>rdfs:label</b>	Yamaha ヤマハ Yamaha Corporation 山葉株式会社
<b>dbpedia-owl:product</b>	dbpedia:Personal_water_craft
<b>dbpprop:numEmployees</b>	26803
<b>dbpedia-owl:locationCountry</b>	dbpedia:Japan
<b>dbpprop:industry</b>	dbpedia:Conglomerate_%28company%29
<b>dbpprop:wordnet_type</b>	<a href="http://www.w3.org/2006/03/wn/w...stances/synset-company-noun-1">http://www.w3.org/2006/03/wn/w...stances/synset-company-noun-1</a>
<b>dbpprop:products</b>	Musical instruments, Audio/Video, Electronics, Computer related products, ATVs, Motorbikes, Vehicle Engines, Personal water craft, golf clubs
<b>dbpprop:hasPhotoCollection</b>	<a href="http://www4.wiwiss.fu-berlin.d...appr/photos/Yamaha_Corporation">http://www4.wiwiss.fu-berlin.d...appr/photos/Yamaha_Corporation</a>
<b>dbpedia-owl:industry</b>	dbpedia:Conglomerate_%28company%29
<b>dbpprop:operatingIncome</b>	140.95 million US\$
<b>dbpprop:companyLogo</b>	dbpedia:YamahaCorp.svg
<b>dbpedia-owl:numberOfStaff</b>	26803
<b>dbpprop:netIncome</b>	million US\$
<b>dbpprop:revenue</b>	4.676 billion US\$
<b>dbpprop:foundation</b>	1887-10-12
<b>foaf:page</b>	<a href="http://en.wikipedia.org/wiki/Yamaha_Corporation">http://en.wikipedia.org/wiki/Yamaha_Corporation</a>
<b>dbpprop:reference</b>	<a href="http://www.yamaha.com">http://www.yamaha.com</a> <a href="http://www.yamaha.com/yamahavg...52526CTID%25253D205200,00.html">http://www.yamaha.com/yamahavg...52526CTID%25253D205200,00.html</a> <a href="http://www.yamaha.com/yasi">http://www.yamaha.com/yasi</a> <a href="http://www.global.yamaha.com/about/brand/index.html">http://www.global.yamaha.com/about/brand/index.html</a> <a href="http://www.yamaha.com/">http://www.yamaha.com/</a>
<b>dbpedia-owl:formationDate</b>	1887-10-12
<b>foaf:depiction</b>	
<b>dbpprop:companyType</b>	dbpedia:Public_company
<b>dbpedia-owl:operatingIncome</b>	1.4095E8

**Fig. 18.8** Facts of Yamaha

The visualizations of the patents described in RDF have links to the WikiSCION and to resources from the Linking Open Data (LOD) cloud which are shown in Fig. 18.6 to Fig. 18.8. These figures show the OpenLink Data Explorer,<sup>8</sup> which is a Firefox browser extension for viewing RDF data.

The rendered RDF content is depicted in Fig. 18.6 showing the metadata for the patent with the title “Pedaling aid for handicapped musician” together with the enriched RDF links to the city of the inventors and to the company which is the assignee of the patent. Figure 18.7 shows how the geographic location of the patent can be used in combination with a map API for displaying the described point of interest. When a user clicks on the “Yamaha Corporation” link and then chooses the describe link in the popup window the browser loads the data on the fly from the Linked Open Data cloud. The result is illustrated in Fig. 18.8 where the company data of Yamaha are highlighted.

## 18.5 Conclusions and Future Work

In this chapter we introduced a novel taxonomy for patent documents: The Wikipedia Science Ontology. This ontology is part of the Patent Taxonomy Integration and Interaction (PTI2) framework that allows for automatic association of

<sup>8</sup><http://ode.openlinksw.com/>.

patent documents with a taxonomy reflecting the “wisdom of crowds”. With PTI2 we are developing an instrument to map the language of IP experts onto the language used in Wikipedia. The interested user of Wikipedia may use PTI2 to retrieve patents relevant to particular Wikipedia pages which complements the manually selected references currently available on these pages.

The current implementation of PTI2 is a first prototype with a number of limitations. These limitations are subject to future research and, after conducting an in-depth evaluation of the system, we will focus on improved strategies for selecting relevant terms for science disciplines and patent documents. Another important topic is the identification of ranking and classification techniques in order to derive the degree of relevance of a patent for a particular scientific discipline. Moreover, we will investigate which paradigms of user interaction are most feasible in the project context. The usage of Linked Open Data for the enrichment of own data sources with external data seems to be very promising. Eventually, the PTI2 framework should become a flexible framework allowing for the integration of different patent ontologies. Consequently a wide variety of interaction methods is enabled ranging from full-text search over concept-based retrieval to hierarchical browsing of patent documents within a single unified interface.

## Appendix: Specific Terms for *Spina bifida*

The following list of terms was automatically generated for the class *Spina bifida* using a document frequency of one:

*abe, amniocentesis, anticonvulsant, arbel, armas, asbah, avrahami, azor, bergh-out, birthmark, blaine, blankets, blues, blurb, bobby, calegory, capra, cerebellum, childbearing, clayden, cordero, cystica, dekel, dimple, drapeau, dursun, dysmorphic, enveloping, erection, fortification, fotheringham, fridman, gros, guinness, guitarist, gwozdz, haemophilia, handicapped, hank, hazneci, hockey, incompletely, intrauterine, iwamoto, izci, jick, kalyon, kibar, kirillova, lipoma, lissauer, lucy, lumbosacral, mcdearmid, melendez, mellencamp, menachem, meningeal, meningocele, merello, mildest, milunsky, moms, mulinare, multivitamins, muraszko, musician, myelomeningocele, myeloschisis, nonimmigrants, numbness, occulta, olympian, ozgul, paralympian, periconceptional, phac, presacral, protrudes, pseudomeningocele, punk, racer, reba, recruits, retroperitoneum, roentgenographic, sacrococcyx, sascha, saxophonist, sbaa, schappell, skateboarder, songwriter, stillbirths, tanni, taskaynatan, torban, tsukimura, unfused, valproic, wakano, wallingford, website-moms, wheelchair, worsening, yoder*

## References

1. Berners-Lee T (2009) Linked data. World Wide Web electronic publication. <http://www.w3.org/DesignIssues/LinkedData>

2. Davies J (2004) Wiki brainstorming and problems with Wiki based collaboration. Tech rep. URL [http://www-users.cs.york.ac.uk/kimble/teaching/students/Jonathan\\_Davies/Jonathan\\_Davies.html](http://www-users.cs.york.ac.uk/kimble/teaching/students/Jonathan_Davies/Jonathan_Davies.html)
3. Fall CJ, Töröcsvári A, Benzineb K, Karetka G (2003) Automated categorization in the international patent classification. SIGIR Forum 37(1):10–25. URL [http://www.acm.org/sigir/forum/S2003/CJF\\_Manuscript\\_sigir.pdf](http://www.acm.org/sigir/forum/S2003/CJF_Manuscript_sigir.pdf)
4. Mukherjea S (2005) Information retrieval and knowledge discovery utilising a biomedical Semantic Web. Brief Bioinform 6(3):252–262. doi:10.1093/bib/6.3.252. URL <http://bib.oxfordjournals.org/cgi/content/abstract/6/3/252>
5. Pesenhofer A, Edler S, Berger H, Dittenbach M (2008) Towards a patent taxonomy integration and interaction framework. In: PaIR'08: Proceeding of the 1st ACM workshop on patent information retrieval. ACM, New York, pp 19–24. URL <http://doi.acm.org/10.1145/1458572.1458578>
6. Precup L, O'Sullivan D, Cormican K, Dooley L (2005) Virtual team environment for collaborative research projects. Int J Innov Learn 3(18):77–94. doi:10.1504/IJIL.2006.008181. URL <http://www.ingentaconnect.com/content/ind/jil/2005/00000003/00000001/art00006>
7. Sah M, Hall W (2007) Building and managing personalized semantic portals. In: WWW'07: Proceedings of the 16th international conference on World Wide Web. ACM, New York, pp 1227–1228. <http://doi.acm.org/10.1145/1242572.1242779>
8. Voss J, Danowski P (2004) Bibliothek, information und dokumentation in der wikipedia. In: Information Wissenschaft und Praxis (8). URL <http://eprints.rclis.org/archive/00002566/>
9. Wanner L, Baeza-Yates R, Brüggemann S, Codina J, Diallo B, Escorsa E, Giereth M, Kompatsiaris Y, Papadopoulos S, Pianta E, Piella G, Puhlmann I, Rao G, Rotard M, Schoester P, Serafini L, Zervaki V (2008) Towards content-oriented patent document processing. World Pat Inf 30(1):21–33. doi:10.1016/j.wpi.2007.03.008
10. Wilkinson DM, Huberman BA (2007) Cooperation and quality in Wikipedia. In: WikiSym'07: Proceedings of the international symposium on Wikis. ACM, New York, pp 157–164. <http://doi.acm.org/10.1145/1296951.1296968>
11. Yu J, Thom JA, Tam A (2007) Ontology evaluation using wikipedia categories for browsing. In: CIKM'07: Proceedings of the sixteenth ACM conference on conference on information and knowledge management. ACM, New York, pp 223–232. <http://doi.acm.org/10.1145/1321440.1321474>
12. Zirn C, Nastase V, Strube M (2008) Distinguishing between instances and classes in the Wikipedia taxonomy. In: Hauswirth M, Koubarakis M, Bechhofer S (eds) Proceedings of the 5th European semantic web conference. LNCS. Springer, Berlin. URL <http://www.eswc2008.org/final-pdfs-for-web-site/onl-4.pdf>
13. W3c sweo linking open data community project (2009) World Wide Web electronic publication. <http://esw.w3.org/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

# Chapter 19

## Automatic Translation of Scholarly Terms into Patent Terms

**Hidetsugu Nanba, Hideaki Kamaya, Toshiyuki Takezawa, Manabu Okumura, Akihiro Shinmori, and Hidekazu Tanigawa**

**Abstract** For a researcher in a field with high industrial relevance, retrieving research papers and patents has become an important aspect of assessing the scope of the field. However, retrieving patents using keywords is a laborious task for researchers, because the terms used in patents (patent terms) are often more abstract than those used in research papers (scholarly terms) or in ordinary language, to try to widen the scope of the claims. We propose a method for translating scholarly terms into patent terms (e.g. translating “word processor” into “document editing device” or “document writing support system”). To translate scholarly terms into patent terms, we propose two methods: the “citation-based method” and the “thesaurus-based method”. We also propose a method combining these two with the existing “Mase’s method”. To confirm the effectiveness of our methods, we conducted some examinations, and found that the combined method performed the best in terms of recall, precision, and  $\epsilon$ , which is an extensional measure of Mean Reciprocal Rank (MRR) widely used for the evaluation of question-answering systems.

---

H. Nanba (✉) · H. Kamaya · T. Takezawa  
Hiroshima City University, 3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima 731-3194, Japan  
e-mail: [nanba@hiroshima-cu.ac.jp](mailto:nanba@hiroshima-cu.ac.jp)

H. Kamaya  
e-mail: [kamaya@ls.info.hiroshima-cu.ac.jp](mailto:kamaya@ls.info.hiroshima-cu.ac.jp)

T. Takezawa  
e-mail: [takezawa@hiroshima-cu.ac.jp](mailto:takezawa@hiroshima-cu.ac.jp)

M. Okumura  
Tokyo Institute of Technology, 4259 Nagatsuta-cho, Midori-ku, Yokohama 226-8503, Japan  
e-mail: [oku@pi.titech.ac.jp](mailto:oku@pi.titech.ac.jp)

A. Shinmori  
INTEC Systems Institute Inc., 1-3-3 Shinsuna, Koto-ku, Tokyo 136-0075, Japan  
e-mail: [shinmori\\_akihiro@intec-si.co.jp](mailto:shinmori_akihiro@intec-si.co.jp)

H. Tanigawa  
IRD Patent Office, 8th floor, OMM Building, 1-7-31, Otemae, Chuo-ku, Osaka 540-0008, Japan  
e-mail: [htanigawa@ird-pat.com](mailto:htanigawa@ird-pat.com)

## 19.1 Introduction

We propose a method for translating scholarly terms<sup>1</sup> into patent terms.<sup>2</sup> For example, our method translates a scholarly term “floppy disc” into patent terms, such as “magnetic recording device” or “removable recording media”. This translation technology can support users when retrieving both research papers and patents.

For a researcher in a field with high industrial relevance, retrieving research papers and patents has become an important aspect of assessing the scope of the field. Examples of such fields are bioscience, medical science, computer science, and materials science. In fact, the development of an information retrieval system for research papers and patents for academic researchers is central to the Intellectual Property Strategic Programs for 2009<sup>3</sup> of the Intellectual Property Strategy Headquarters in the Cabinet Office, Japan. In addition, research paper searches and patent searches are required by examiners in government patent offices, and by the intellectual property divisions of private companies. An example is the execution of an invalidity search among existing patents or research papers that could invalidate a rival company’s patents or patents under application in a Patent Office. However, the terms used in patents are often more abstract or creative than those used in research papers or in ordinary language [1], to try to widen the scope of the claims. Therefore, a technology for translating scholarly terms into patent terms is required.

This technology is also useful in the following situation. When inventors or patent agents write patents, they are often confused about which patent terms they should use, because there may be several choices of patent terms for a scholarly term. For example, the scholarly term “floppy disc” can be expressed as “removable recording medium”, if the inventors or patent attorneys focus on the floppy disc’s feature of removability. On the other hand, “floppy disc” can also be expressed as “magnetic recording medium”, if they focus on the feature of recording information using magnetic force. In such a situation, if it can generate a list of candidate patent terms for a given scholarly term, this technology would support the inventors and the patent attorneys while writing patents.

The remainder of this paper is organised as follows. Section 19.2 explains the behaviour of our system. Section 19.3 describes some related work. Section 19.4 proposes our method for translating scholarly terms into patent terms. Section 19.5 discusses how we investigated the effectiveness of our method by conducting some examinations, and discusses our experimental results. Finally, we provide our conclusions in Sect. 19.6.

---

<sup>1</sup>Generally, technical terms are defined as terms used in a particular research field. Based on this definition, “floppy disc” or “word processor” are not technical terms, because they are commonly used. In this paper, we define “scholarly terms” as terms used in research papers, even though they may also be used more generally, such as “floppy disc” or “word processor”.

<sup>2</sup>We define the task of “translation of scholarly terms into patent terms” as “to output all useful patent terms for patent retrieval”. In many cases, patent terms are hypernyms or synonyms of a given scholarly term, and include a part of scholarly terms.

<sup>3</sup>[http://www.kantei.go.jp/jp/singi/titeki2/keikaku2009\\_e.pdf](http://www.kantei.go.jp/jp/singi/titeki2/keikaku2009_e.pdf). Cited 30 June 2010.

## 19.2 System Behaviour

In this section, we describe our system that can retrieve both research papers and patents using the function of automatic translation of scholarly terms into patent terms. In Fig. 19.1, search forms on the left and right sides are for research papers and for patents, respectively. In the following, we explain a search procedure for users, who are unfamiliar with patent searches. The procedure consists of the following three steps.

- **(Step 1)** Input a scholarly term in the “Title” field (shown as (1) in Fig. 19.1) in a search form for research papers.<sup>4</sup>
- **(Step 2)** Click the “related patent terms” button (shown as (2)), then some candidate patent terms, such as “storage medium”, “recording medium”, and “disc recording medium”, are shown in a pop-up window (shown as (3)).
- **(Step 3)** Select appropriate candidates by checking the related boxes, then they are automatically inserted into the “Title of Invention” field (shown as (4)) in a search form for patents.

If users do not have enough knowledge or skills for a patent search, it is difficult for them to conceive of appropriate patent terms. However it is possible for them to select appropriate patent terms from a list of candidates. We propose a method that translates scholarly terms into patent terms, and experimentally confirm its effectiveness.

## 19.3 Related Work

An invalidity search task was performed in the Patent Retrieval Task of the Fourth [3], the Fifth [4], and the Sixth [5] NII Test Collection for Information Retrieval (NTCIR) workshops. The goal of this task was to retrieve patents that could invalidate existing claims. Five groups with 21 systems participated in the Japanese retrieval subtask in the Sixth NTCIR, and the systems were evaluated using the Mean Average Precision (MAP). The best system obtained a MAP of 0.0815 [12]. The system analysed the structure of queries, and weighted terms in particular essential parts of the queries, using several weighted methods, such as the inverse document frequency (idf) without a term frequency (tf) method. In contrast to this task, we aimed to construct a system retrieving not only patents but also research papers that could invalidate existing claims.

There has been much research in the field of cross-genre information access, such as that discussed in the technical survey task of the Patent Retrieval Task of the Third NTCIR workshop [9]. This task aimed to retrieve patents relevant to a given newspaper article. In this task, Itoh et al. focused on “Term Distillation” [8]. The distribution of the frequency of the occurrence of words was considered to be different between heterogeneous databases. Therefore, unimportant words were assigned

---

<sup>4</sup>In this case, the scholarly term “floppy disc” was already inserted in the “title” field.



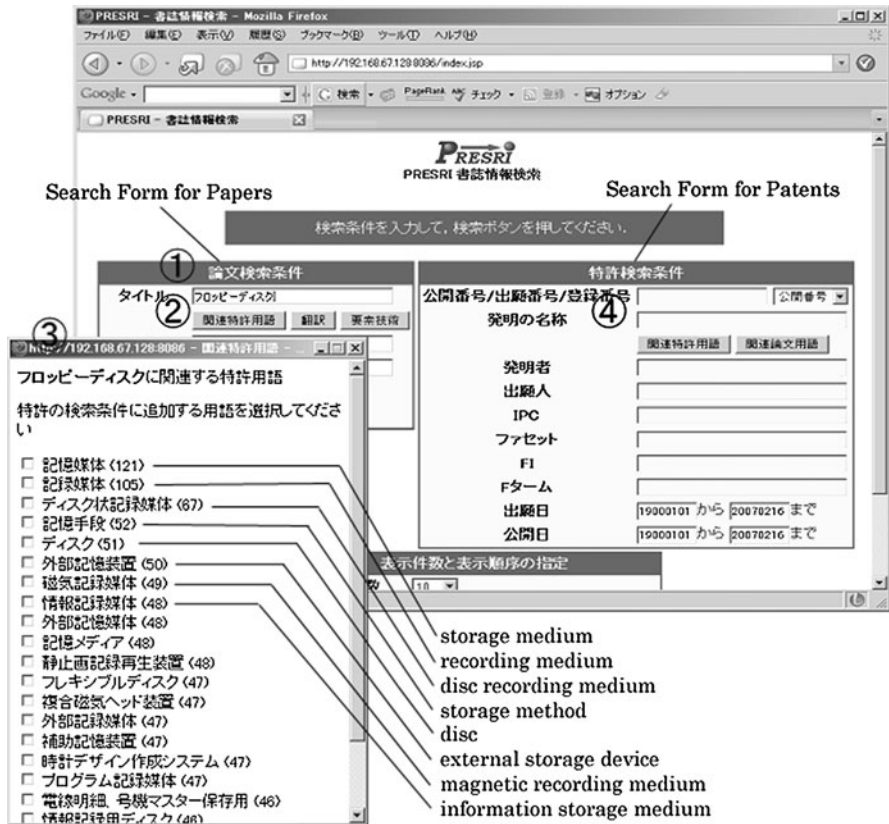


Fig. 19.1 System snapshot

high scores when using tf\*idf to weight words. Term Distillation is a technique that can prevent such cases by filtering out words that can be assigned incorrect weights. This idea was also used to link news articles and blog entries [7]. However, some patent terms, such as “magnetic recording device”, appear only in a patent database, and “Term Distillation” cannot be applied in such cases.

There are another several research projects related to cross-genre information access [10, 15, 16]. TREC Chemical IR Track aimed for cross-genre information retrieval using research papers and patents in the chemical field.

The Patent Mining Task in the Seventh and the Eighth NTCIR workshop [15, 16] aimed to create technical trend maps from a set of research papers and patents. The following two subtasks were conducted in this task.

1. Subtask of research papers classification: Classification of research papers into the International Patent Classification (IPC) system.
2. Subtask of technical trend map creation: Extraction of expressions of elemental technologies and their effects from research papers and patents.

However, there were no participant groups that translate scholarly terms into patent terms.

As another approach for cross-genre information access, Nanba et al. proposed a method to integrate a research paper database and a patent database by analysing citation relations between research papers and patents [14]. For the integration, they extracted bibliographic information of cited literatures in “prior art” fields in Japanese patent applications. Using this integrated database, users can retrieve patents that relate to a particular research paper by tracing citation relations between research papers and patents. However, the number of cited papers among patent applications is not enough to retrieve related papers or patents, even though the number of opportunities for citing papers in patents or for citing patents in papers has been increasing recently. We therefore have studied automatic translation of scholarly terms into patent terms.

Chen et al. also addressed the vocabulary mismatch problem [2]. Different terms in different domains having the same concept prevent us from conducting cross-domain retrieval. For the problem, they firstly created different thesauri from research papers in two biological sub domains and then associated pairs of terms in them. Although we could not examine their method due to the unavailability of a large-scale Japanese research paper database, it is worth to consider as one of our future works.

## **19.4 Automatic Translation of Scholarly Terms into Patent Terms**

We propose three translation methods: the “citation-based method”, the “thesaurus-based method”, and “Mase’s method”. We describe these methods in the following subsections. We then describe a method that combines the three methods.

### ***19.4.1 Translation Using Citation Relationships Between Research Papers and Patents***

A research paper and a patent that have citation relationships with each other, generally tend to be in the same research field. Using this idea, translation of a scholarly term can be realised by using the following procedure.

1. Input a scholarly term.
2. Retrieve research papers that contain the given scholarly term in their titles.
3. Collect patents that have citation relationships with the papers retrieved in Step 2.
4. Extract patent terms from patents collected in Step 3, and output them in order of frequency.

We call this the “citation-based method”. To extract patent terms from patents that were collected in Step 3, we focused on the patent claims, which are considered

(original)

車体に固定する筐体内に前後揺動体を車体前後方向へ回動可能に軸支し、該前後揺動体に揺動基部を車体左右方向へ回動可能に軸支し、該揺動基部に植設したシフトレバーを車体前後及び左右方向へ揺動させることにより筐体上面に形成したゲート部を移動し所望のレンジを選択して自動変速機を切替操作する シフトレバー装置 において、前記揺動基部に一對の上下に離間した突起部を設け、一方の突起部に当接可能なP係止部と、他方の突起部に当接可能なN係止部を有する回転ロック体を前記筐体に回動可能に軸支するとともに、シフトレバーがPレンジ又はNレンジに移動したとき該回転ロック体を回動させるアクチュエータを前記筐体に固定したことを特徴とする シフトレバー の シフトロック装置。

(translation)

A shift lock unit for a shift lever device for use in an automatic transmission having parking and neutral ranges, comprising: a casing; a base rotatably supported in said casing, said base including first and second protrusions separate d from each other in a longitudinal direction thereof; a lock member rotatably supported in said casing, said lock member including first and second engagements which can abut on said first and second protrusions, respectively; a shift lever arranged with said base, said shift lever serving to select a desired range of the transmission; and an actuator fixed to said casing, said actuator rotating said lock member when said shift lever is moved to one of the parking and neutral ranges.

**Fig. 19.2** A sample Japanese claim extracted from a patent (publication number = 10-184868)

the most important part of each patent. A patent claim is the precise legal definition of the invention, identifying the specific elements of the invention for which the inventor is claiming rights and seeking protection.

In most claims, noun phrases before “において” (concerning) and after “を特徴とする” (characterised by) indicate topic terms in patents [17]. Figure 19.2 is an example of a Japanese sample claim. In this figure, the underlined terms “シフトレバー装置” (shift lever device), “シフトレバー” (shift lever), and “シフトロック装置” (shift lock unit) are extracted as topic terms.

### 19.4.2 Translation Using an Automatically Constructed Thesaurus

To enlarge the scope of the patent, hypernyms of scholarly terms are often used in patents. We therefore propose a method using a thesaurus in addition to the citation-based method. We used a hypernym/hyponym thesaurus, which Nanba [13] automatically constructed using a pattern “A や B などの C” (C, such as A (or | and) B) [6]. The thesaurus contains 7,031,159 hypernym/hyponym relations, which were extracted from Japanese patents published in the 10 years from 1993 to 2002. This thesaurus also give the frequencies of each hypernym/hyponym relation in patents.

Using this thesaurus, we realise translation of a scholarly term by extracting hypernyms of the given scholarly term from the thesaurus,<sup>5</sup> and by outputting them in order of frequency. We call this the “thesaurus-based method”.

<sup>5</sup>For example, when a scholarly term “floppy disc” is given, the thesaurus-based method output its hypernyms, such as “removable recording medium”, as patent terms.

### 19.4.3 Translation Using Mase's Method

In patent applications, inventors may explicitly describe related terms by using parentheses, as in “floppy disc (magnetic recording medium)”. The term preceding the parentheses and the term in parentheses have a broader/narrower relationship. Mase et al. [11] extracted related term from the text in the “description of symbols” fields of Japanese patents. They experimentally confirmed that these terms are effective for query expansion of patent retrieval. This method can also be used in our work.

Using Mase's method, we realise translation of a scholarly term by extracting related terms of a given scholarly term from the “description of symbols” fields, and by outputting them in order of frequency.

### 19.4.4 Translation Combining the Three Methods

We propose a method combining the above three methods in the following two steps.

#### 19.4.4.1 (Step 1) Combining Mase's Method with the Other Two Methods

Using Mase's method, we extracted a total of 679,931 pairs of related terms. We translated some scholarly terms into patent terms and found that Mase's method could output correct patent terms at high rates. However, the number of terms obtained by Mase's method is very small and in the worst case, no terms were output.<sup>6</sup> Therefore, we improve the citation-based and the thesaurus-based methods using Mase's method. Consider an example in which Mase's method obtained two patent terms “magnetic recording device” and “removable storage device” for a given scholarly term “floppy disc”. From these results, “floppy disc” can be inferred to be a term related to a “device”, because the last word of both patent terms is “device”. If there is another patent term for “floppy disc”, the last word of the term is probably “device”. Therefore, we improve both the citation-based and the thesaurus-based methods by giving a higher priority using Mase's method. The procedure is as follows.

1. Normalise the scores (frequencies) of each candidate term in a list given by the citation-based method (or the thesaurus-based method) to a value between 0 and 1 by dividing each score by the score of the term ranked 1.
2. Extract the last word of each candidate term obtained by Mase's method. In this step, we also extract the frequencies of each term in “description of symbols” fields.<sup>7</sup>

---

<sup>6</sup>We will report this experimental result later.

<sup>7</sup>When Mase's method outputs three candidate terms “magnetic recording device” (freq. 10), “removable storage device” (freq. 5), and “information recording medium” (freq. 3), the three words “device” (freq. 10), “device” (freq. 5), and “medium” (freq. 3) are extracted from the terms.

3. Sum the scores (frequencies) for each last word obtained in Step 2, and normalise them to a value between 0 and 1 by dividing each score by the score of the word at rank 1.<sup>8</sup>
4. When the last word of a candidate term by the citation-based method (or the thesaurus-based method) and one of the words obtained in Step 3 match, give the scores of their words to each term, and output in order of score.<sup>9</sup>

#### 19.4.4.2 (Step 2) Combining the Citation-Based Method and the Thesaurus-Based Method

The terms output by both the citation-based and the thesaurus-based methods, which were improved by Mase's method, seem to be correct patent terms. We therefore combine both methods using the following equation.

$$\begin{aligned} &\text{Score of a candidate patent term by the combined method} \\ &= \lambda * \text{Score by the citation-based method} \\ &\quad + (1 - \lambda) * \text{Score by the thesaurus-based method} \end{aligned}$$

Here,  $\lambda$  is a parameter that adjusts the effects of the citation-based and the thesaurus-based methods. We will describe how to determine this parameter in Sect. 19.5.1.

## 19.5 Experiments

To confirm the effectiveness of our methods, we conducted some examinations. We describe the experimental conditions in Sect. 19.5.1, report the experimental results in Sect. 19.5.2, and discuss the results in Sect. 19.5.3.

### 19.5.1 Experimental Conditions

#### 19.5.1.1 Documents

We used Japanese patent applications published in the 10 years from 1993 to 2002. We also used about 85,000 bibliographic records of cited papers in patents, which were automatically created using Nanba's method [14].

We created the correct data set using the following procedure.

---

<sup>8</sup>For the example in Step 2, "device" (score 15) and "medium" (score 3) are obtained. Then, the scores of the words are normalised by dividing by 15, which is the score for "device", resulting in "device" (score 1) and "medium" (score 0.2).

<sup>9</sup>For example, if the citation-based method obtained a term "recording medium" (score 0.5), a score  $0.2 \times m$  for "medium" is added to 0.5. Here,  $m$  is a parameter that indicates the influence of Mase's method on the citation-based method. We will describe how to determine  $m$  in Sect. 19.5.1.

1. Extract all noun phrases from the 85,000 bibliographic records of cited papers in patents, and rank them in order of frequency.
2. Manually select scholarly terms from the noun phrases.
3. Output candidate terms using all our methods and baseline methods, which we will describe later.
4. Manually identify correct patent terms in all candidates obtained in Step 3.

For the identification of correct terms in Step 4, we used the following four criteria.

- If candidate terms are components of a given scholarly term, we identified the candidates as incorrect. For example, a patent term “document editing system” is correct for the scholarly term “word processor”, while the term “display system” is incorrect, because a display system is a component of a word processor.
- When the frequency of a candidate term in a patent database was very low, we identified the term as incorrect, because it is not a common expression in patents, and is therefore not useful for patent searches.
- When a candidate term is too abstract in comparison with a scholarly term, we identified it as incorrect. For example, the candidate “magnetic recording device” is correct for a scholarly term “floppy disc”, while “information storage system” is incorrect, because the term is too abstract and has many hyponyms.
- When a patent term is spelled in several different ways, such as “disc recording medium” or “disk recording medium”, we identified them as correct.

Finally, we obtained 47 scholarly terms (input) with 2.8 patent terms (output) on average for each scholarly term. We show some of these in Table 19.1.

We investigated whether these 47 terms were not skewed to particular fields. We retrieved patents by an online patent retrieval system “Kantan Tokkyo Kensaku”<sup>10</sup> using these terms as keywords. Then, we extracted categories (IPC codes at the subclass level) having the largest number of retrieved patents for each keyword. The results are shown in Table 19.2. Taking account of class imbalance of IPC taxonomy and skew of the distribution of IPC relevant to academic fields, it is considered that the results in Table 19.2 are within the allowable range.

### 19.5.1.2 Evaluation Measure

As an evaluation measure, we used  $\epsilon$ , which is an expansion of MRR, a standard evaluation measure for evaluating question-answering systems. The evaluation score will be close to 1 when many correct terms are given high ranks.

$$\epsilon = \frac{\sum_{i \in R} \frac{1}{i}}{\sum_{j \in \{1, 2, \dots, n\}} \frac{1}{j}}.$$

---

<sup>10</sup><http://kantan.nexp.jp/>.

**Table 19.1** Data for evaluation (example)

Scholarly term (input)	Patent term (output)
DRAM	semiconductor memory, dynamic memory, dynamic random access memory
memory cell	semiconductor memory device
word processor	document processing device, document information processing device, document editing system, document writing support system
TV camera	photographic device, image shooting apparatus, image pickup apparatus

Here,  $n$  indicates the number of correct patent terms for a given scholarly term,  $R$  indicates a set of ranks of correct terms in a system output,  $i$  is the rank of a correct term in a system output. In addition to the  $\epsilon$  measure, we also used recall and precision.

$$\text{Recall} = \frac{\text{The number of correctly extracted patent terms}}{\text{The number of correct patent terms}},$$

$$\text{Precision} = \frac{\text{The number of correctly extracted patent terms}}{\text{The number of candidate terms extracted by a system}}.$$

We evaluated only the top 20 terms in each system output.

### 19.5.1.3 Alternatives

We conducted experiments using the following nine methods. Abbreviations for each method are shown in parentheses.

#### *Our methods*

- (1) Citation-based method (Cite)
- (2) (1) + improvement by Mase's method (Cite(M))
- (3) Thesaurus-based method (Thes)
- (4) (3) + improvement by Mase's method (Thes(M))
- (5) (2) + (4) combined method (Cite(M) + Thes(M))

#### *Baseline methods*

- (6) Mase's method (Mase)
- (7) Term co-occurrence-based method (GETA)
- (8) Synonyms extraction method (Syn)
- (9) Japan Science and Technology thesaurus-based method (JST)

Methods (1), (3), (5), and (6) correspond to those mentioned in Sects. 19.4.1, 19.4.2, 19.4.3, and 19.4.4, respectively. Methods (2) and (4) are improved by Mase's

**Table 19.2** Distribution of data for evaluation

IPC	Frequency	Explanation of IPC
H01L	8	semiconductor devices, electric solid state devices
C12N	7	micro-organisms or enzymes
G11B	7	information storage based on relative movement between record carrier and transducer
G06F	6	optical computing devices
A63F	2	card, board, or roulette games, indoor games using small moving playing bodies
G09G	2	arrangements or circuits for control of indicating devices
G02B	2	optical elements, systems, or apparatus
B41M	1	printing duplicating, marking, or copying processes, colour printing
A61B	1	diagnosis, surgery, identification
G03F	1	photomechanical production of textured or patterned surfaces
H01M	1	processes or means for the direct conversion of chemical energy into electrical energy
C09D	1	coating compositions, filling pastes, chemical paint or ink removers, inks correcting fluids
H05K	1	printed circuits, casings or constructional details of electric apparatus
G01N	1	investigating or analysing materials by determining their chemical or physical properties
G06Q	1	data processing systems or methods
H04N	1	pictorial communication
H03H	1	impedance network, resonators
H01S	1	devices using stimulated emission
G03B	1	apparatus or arrangements for taking photographs or for projecting or viewing them
B41J	1	typewriters, selective printing mechanisms, correction of typographical errors

method as described in Sect. 19.4.3. We will explain the procedures for parameter tuning later.

As one baseline method, we employed the word co-occurrence method (7). In this method, terms co-occurred frequency with a given scholarly term are extracted as candidates using the IR engine GETA.<sup>11</sup>

As another baseline method, we used an automatically constructed synonym dictionary [13]. Nanba constructed a thesaurus using a pattern “A や B などの C”

<sup>11</sup><http://geta.ex.nii.ac.jp/>.



(C, such as A (or | and) B). In the expressions, there are several cases in which parentheses were used. Following is an example that matches the pattern.

A flexible inner tube 21 containing synthetic resin, such as *PTFE* (*polytetrafluoroethylene*), is inserted through an outer tube.

From the expression, Nanba extracted “polytetrafluoroethylene” as a synonym of “PTFE”. He obtained 50,161 pairs of synonyms, and confirmed that the synonyms were useful for query expansion in patent retrieval. We used this synonym dictionary as a baseline method (8).

As the other baseline method, we used a free online thesaurus (JST thesaurus),<sup>12</sup> which was provided by the Japan Science and Technology Agency (JST). Using the JST thesaurus, we translate scholarly terms into patent terms, in the same way as the thesaurus-based method, which we mentioned in Sect. 19.4.2.

#### 19.5.1.4 Parameters in Methods (2), (4), and (5)

We conducted a pilot study to determine a value for parameter  $m$ , which indicates the influence of Mase’s method for both the citation-based and the thesaurus-based methods, and a value of  $\lambda$ , which adjusts the relative contributions of the citation-based and the thesaurus-based methods. We prepared a data set that consists of 25 scholarly terms and their correct patent terms, and used it for the pilot study. The pilot study was conducted in two steps. In the first step, we changed values of  $m$  from 0 to 1 at 0.1 intervals, and calculated  $\epsilon$  scores of the citation-based method (2) and the thesaurus-based method (4). We found the highest  $\epsilon$  scores, when  $m$  for method (2) was 0.8, and  $m$  for method (4) was 0.2 (Fig. 19.3). In the second step, we optimised the  $\lambda$  score by changing it from 0 to 1 at 0.1 intervals, and calculating the  $\epsilon$  scores for each step. We obtained the highest  $\epsilon$  score, when  $\lambda$  was 0.3. We used this score for the combined method (9).

### 19.5.2 Experimental Results

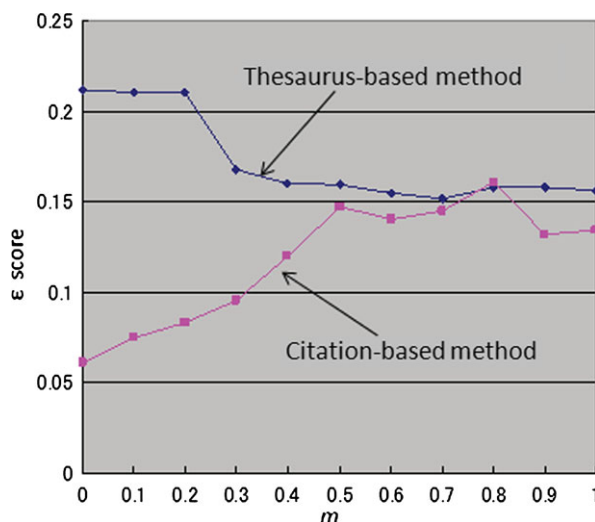
The  $\epsilon$  scores for each method are shown in Table 19.3. We further investigated our methods (2), (4), (5) and baseline methods (6) and (8), all of which obtained better scores among all methods compared. The results are shown in Table 19.4. In the table, we also show recall, precision, and  $\epsilon$  scores for an ideal system. Here, precision scores for the ideal system were less than 1, because the average number of correct patent terms for each scholarly term is 2.8. Therefore, the scores for the ideal system are an upper bound.

In Table 19.3, we see that the  $\epsilon$  score for method (2) is larger by 0.037 points than that by for method (1), which indicates that Mase’s method was effective in

---

<sup>12</sup>[http://jdream2.jst.go.jp/html/thesaurus99/thesaurus\\_index99.htm](http://jdream2.jst.go.jp/html/thesaurus99/thesaurus_index99.htm).

**Fig. 19.3** Determination of a value for parameter  $m$



improving the citation-based method. On the other hand, Mase's method did not improve the thesaurus-based method, because the difference in  $\epsilon$  scores for methods (3) and (4) is only 0.009. However, the performance by the thesaurus-based method was good enough, and there was little room to improve the thesaurus-based method by Mase's method. The combined method (5) obtained the best  $\epsilon$  score of all methods. This method also obtained best recall and precision scores in Table 19.4.

In Table 19.3, the  $\epsilon$  score for the JST thesaurus-based method (9) was smaller than those for the thesaurus-based methods (3) and (4), although the JST thesaurus was manually created, while the thesaurus used in methods (3) and (4) was created automatically. This result was caused by the number of terms in the JST thesaurus. The original JST thesaurus contains about 400,000 scholarly terms, but the freely available online version contains only 10% of the original. As a result, there were many cases in which no terms were extracted by the method (9). If we had been able to use the original one, the performance of method (9) would be better.

**Table 19.3** Evaluation using  $\epsilon$

Our method					Baseline			
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Cite	Cite(M)	Thes	Thes(M)	Cite(M)+Thes(M)	Mase	GETA	Syn	JST
0.136	0.173	0.231	0.240	<b>0.298</b>	0.107	0.011	0.058	0.050

**Table 19.4** Evaluation using  $\epsilon$ , Recall, and Precision

Method		Measure	top 5	top 10	top 15	top 20
Our method	(2) Cite(M)	$\epsilon$	0.151	0.165	0.170	0.173
		Recall	0.169	0.242	0.275	0.311
		Precision	0.115	0.073	0.056	0.047
	(4) Thes(M)	$\epsilon$	0.213	0.235	0.239	0.240
		Recall	0.274	0.362	0.393	0.399
		Prec.	0.145	0.104	0.078	0.061
	(5) Cite(M)+Thes(M)	$\epsilon$	<b>0.261</b>	<b>0.286</b>	<b>0.292</b>	<b>0.298</b>
		Recall	<b>0.309</b>	<b>0.421</b>	<b>0.459</b>	<b>0.533</b>
		Precision	<b>0.170</b>	<b>0.121</b>	<b>0.092</b>	<b>0.076</b>
Base-line	(6) Mase	$\epsilon$	0.083	0.097	0.106	0.107
		Recall	0.108	0.172	0.246	0.264
		Precision	0.072	0.061	0.055	0.045
	(8) Syn	$\epsilon$	0.054	0.055	0.057	0.058
		Recall	0.080	0.087	0.101	0.104
		Precision	0.053	0.038	0.037	0.035
	Upper bound	$\epsilon$	1.000	1.000	1.000	1.000
		Recall	1.000	1.000	1.000	1.000
		Precision	0.587	0.294	0.196	0.147

### 19.5.3 Discussion

#### 19.5.3.1 Comparison of the Citation-Based Method (2), the Thesaurus-Based Method (4), and the Combined Method (5)

In Table 19.3, we see that the combined method's (5)  $\epsilon$  score was improved to 0.298 from the thesaurus-based method's (4) score of 0.240. Furthermore, method (5) had a significantly improved recall score in comparison with method (4) from 0.399 to 0.533. To investigate the reasons for significant improvement of the recall score by method (5), we counted the number of cases for which method (5) was better an method (4) in recall, precision, and  $\epsilon$ . We show the results in Table 19.5. In the table, for example, "C+T(5) < Thes(4)" in the first column indicates "the score for method (5) is smaller than that for method (4)", and the numbers of such cases for recall, precision, and  $\epsilon$  are shown in the second, third, and fourth columns, respectively. The results showed that the number of cases in which the combined method (5) impaired both recall and precision scores compared with method (4) is two (4.3%), while the number of cases of improvement is 13 (27.6%). In the same way, we also compared method (5) and the citation-based method (2). These results also showed that method (5) could significantly improve method (2) (Table 19.5).

**Table 19.5** Comparison of system outputs by the citation-based method (2), the thesaurus-based method (4), and the combined method (5)

	Recall	Precision	$\epsilon$
C+T(5) < Thes(4)	2	2	14
C+T(5) > Thes(4)	13	13	17
C+T(5) = Thes(4)	31	31	16
C+T(5) < Cite(2)	3	4	9
C+T(5) > Cite(2)	23	23	25
C+T(5) = Cite(2)	21	20	13

**Table 19.6** The proportion of cases for which each system could not correctly convert scholarly terms within the top 20

Our methods			Baseline methods	
(2) Cite(M)	(4) Thes(M)	(5) Cite(M)+Thes(M)	(6) Mase	(8) Syn
48.9% (23/47)	40.4% (19/47)	25.5% (12/47)	55.3% (26/47)	74.5% (35/47)

From these experimental results, we can conclude that our combined method (5) is valid.

### 19.5.3.2 Recall Scores of Each Method

In the results in Table 19.5, we found many cases for which both recall and precision scores for the methods (2) and (4) were the same as those for method (5). We investigated these cases, and found that there were no correct patent terms within the top 20 of the system output. In such cases, the system could not support users in real situations of patent searches. We therefore counted the number of such cases. The results are shown in Table 19.6. From the results, we see that method (2) could not extract correct patent terms within the top 20, and the baseline methods could not extract terms for more than half of the cases. On the other hand, the number of cases for which method (5) could not output correct patent terms within the top 20 was only 12 (25.5%), so method (5) is useful in 3/4 of all cases. Five of these 12 were “asparagine acid”, “trehalose”, “carboxyl group”, “lithium niobate”, “dimethyl sulfoxide”, “novolac resin”, all of which were in the chemical domain. Although there are not enough cases to be sure, the results indicate that the performance of our methods might vary with the research field.

## 19.6 Conclusions

In this paper, we have proposed three methods: the citation-based method, the thesaurus-based method, and the method combining these two methods. To con-

firm the effectiveness of our methods, we conducted some examinations. We found that the combined method performed the best in terms of recall, precision, and  $\epsilon$ , which is an extensional measure of Mean Reciprocal Rank (MRR) widely used for the evaluation of question-answering systems. In 25.5% of cases, the combined method could not extract correct patent terms within the top 20, which is a smaller proportion than that found for other methods.

## References

1. Atkinson KH (2008) Toward a more rational patent search paradigm. In: Proceedings of the 1st international CIKM workshop on patent information retrieval (PaIR'08), pp 37–40
2. Chen H, Ng TD, Martinez J, Schats BR (1997) A concept space approach to addressing the vocabulary problem in scientific information retrieval: An experiment on the worm community system. *J Am Soc Inf Sci* 48(1):17–31
3. Fujii A, Iwayama M, Kando N (2004) Overview of patent retrieval task at NTCIR-4. Working notes of the 4th NTCIR workshop, pp 225–232
4. Fujii A, Iwayama M, Kando N (2005) Overview of patent retrieval task at NTCIR-5. In: Proceedings of the 5th NTCIR workshop meeting on evaluation of information access technologies: Information retrieval, question answering and cross-lingual information access, pp 269–277
5. Fujii A, Iwayama M, Kando N (2007) Overview of the patent retrieval task at NTCIR-6 workshop. In: Proceedings of the 6th NTCIR workshop meeting, pp 359–365
6. Hearst MA (1992) Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th international conference on computational linguistics, pp 539–545
7. Ikeda D, Fujiki T, Okumura M (2006) Automatically linking news articles to blog entries. In: Proceedings of AAAI spring symposium series computational approaches to analyzing weblogs, pp 78–82
8. Itoh H, Mano H, Ogawa Y (2002) Term distillation for cross-db retrieval. Working notes of the 3rd NTCIR workshop meeting, Part III: Patent retrieval task, pp 11–14
9. Iwayama M, Fujii A, Kando N, Takano A (2002) Overview of patent retrieval task at NTCIR-3. Working notes of the 3rd NTCIR workshop meeting, Part III: patent retrieval task, pp 1–10
10. Lupu M, Piroi F, Huang J, Zhu J, Tait J (2009) Overview of the TREC chemical IR track. In: Proceedings of the 18th text retrieval conference
11. Mase H, Matsubayashi T, Ogawa Y, Yayoi Y, Sato Y, Iwayama M (2005) NTCIR-5 patent retrieval experiments at Hitachi. In: Proceedings of NTCIR-5 workshop meeting, pp 318–323
12. Mase H, Iwayama M (2007) NTCIR-6 patent retrieval experiments at Hitachi. In: Proceedings of the 6th NTCIR workshop meeting, pp 403–406
13. Nanba H (2007) Query expansion using an automatically constructed thesaurus. In: Proceedings of the 6th NTCIR workshop meeting, pp 414–419
14. Nanba H, Anzen N, Okumura M (2008) Automatic extraction of citation information in Japanese patent applications. *Int J Digit Libr* 9(2):151–161
15. Nanba H, Fujii A, Iwayama M, Hashimoto Y (2008) The patent mining task in the seventh NTCIR workshop. In: Proceedings of the 1st international CIKM workshop on patent information retrieval (PaIR'08), pp 25–31
16. Nanba H, Fujii A, Iwayama M, Hashimoto T (2010) Overview of the patent mining task at the NTCIR-8 workshop. In: Proceedings of the 8th NTCIR workshop meeting on evaluation of information access technologies: Information retrieval, question answering and cross-lingual information access, pp 293–302
17. Shinmori A, Okumura M, Marukawa Y, Iwayama M (2002) Rhetorical structure analysis of Japanese patent claims using cue phrases. In: Proceedings of the 3rd NTCIR workshop meeting, Part III: patent retrieval task, pp 69–77

# Chapter 20

## Future Patent Search

John I. Tait and Barou Diallo

**Abstract** In this chapter we make some prediction for patent search in about ten year's time—in 2021. We base these predictions on both the contents of the earlier part of the book, and on some data and trends not well represented in the book (for one reason or another). We consider primarily incorporating knowledge of different sorts of patent search into the patent search process; utilising knowledge of the subject domain of the search into the patent search system; utilising multiple sources of data within the search system; the need to address the requirement to deal with multiple languages in patent search; and the need to provide effective visualisation of the results of patent searches. We conclude the real need is to find ways to support search independent of language or location.

### 20.1 Introduction

In this chapter we will try to move from the reviews of current professional practice in patent search and of recent relevant scientific research to try to foresee how the field will develop in the medium-term future. In particular we will try to see how changes in available technology will impact the tools available to patent professionals; and the issues and pressures, which will inhibit or accelerate these changes.

One of the characteristics of patent search by professional searchers outside the patent offices is that it is usually driven by briefs from strategic level business managers or by patent attorneys, and it aims to produce a report summarising the facts about a particular area (for example whether a particular device may infringe one or more existent patents, or the likelihood that one could in principle produce a new substance of a particular type which neither infringes existing patents, nor has been previously published, and therefore may be legitimately patented). Present day sys-

---

J.I. Tait (✉)

Information Retrieval Facility, Techgate, Donau City Strasse 1, Vienna, 1220, Austria  
e-mail: [john.tait@ir-facility.org](mailto:john.tait@ir-facility.org)

B. Diallo

European Patent Office, Patentlaan 2, 2288 EE Rijswijk Zh, Netherlands  
e-mail: [bdiallo@epo.org](mailto:bdiallo@epo.org)

tems rely on the searcher bringing together the data from several of sources (for example patent databases, academic journal collections and legal information sources) in a coherent whole in an essentially unsupported manner: future systems will explicitly provide support.

Perhaps the biggest challenge here is integrating the search systems with corporate and enterprise information systems (whether science and technology support or business systems), especially given the move of patent search from being an in-house operation to being outsourced. Generalised outsourcing operations are unlikely to have access to the quality of information in-house operations would have. Further they are unlikely to be given access to the full range of confidential information available in house.

Of course this focusses on patent search from technology user point of view, whether it be the inventor, companies seeking freedom to operate, or strategic business managers. Another important group of users of patent search systems is the patent offices themselves. Necessarily a patent examiner in a patent office has different needs from a commercial searcher, and future integrated systems, which better reflect searchers needs, and workflows will inevitably therefore be different to some degree. More automated, information-rich, and task aware patent search systems can help patent offices achieve a particular implicit goal of the patent offices: to improve both the volume of applications dealt with by an examiner, whilst simultaneously improving the quality of the granted patents. This is certainly an effective way to address THE key challenge facing patent offices: reducing the time between application and grant or rejection whilst improving the quality and in particular the defensibility of the granted patents.

The IP community see the principal issues to be the backlog of unexamined applications and the costs associated with the granting process. But there is little real benefit in rapidly obtaining a patent which cannot be enforced in court, or which promotes lengthy and complex litigation. This will not be in the interests of inventors, patent holders, or intending technology exploiters. Better technological support for the search process can allow quality to be improved whilst reducing time-to-grant and human effort in review.

Further, many patents currently granted are of suspect validity because of the weakness of current practices in searching non-English patent data (disproportionately growing as a proportion of the whole, especially in Asia). This leaves aside the rarity of searching of non-English non-patent public data for evidence of lack of novelty.

Having the possibility to obtain information is not equivalent to reviewing and understanding the underlying information. As discussed elsewhere in this volume, the reasons for performing a patent search are multiple. The most obvious is to determine whether or not an applicant can get a patent or if its invention has already been patented. Other reasons might include:

- Getting an idea of how an application and patent is drafted to help in the preparation of a new application
- Learning more about a new technical field
- For competitive market information and tracking

## 20.2 Patent Search in 2021

In ten years time we would expect to see patent search systems with a number of characteristics not present in today's search systems.

- First, there will be a reflection of the different sorts of patent search task, their differing characteristics and integrated tools which reflect the domain: chemical versus mechanical versus telecoms and so on, and event-driven legal status searches versus document content driven searches.
- Second, systems will provide access to both patent and non-patent literature in a single integrated environment. There will also be a firm separation between data and the systems used to access it—so that several data sources (we avoid the term databases) can be accessed by the searcher within a single search and analysis environment.
- Third, the tools will be inherently multi-lingual—allowing the English speaking patent searcher to deal with Chinese data more or less as easily as with English data, for example.
- Fourth, complex visualisations will be provided to support not only specialist tasks like technology landscape mapping, but also to help the searcher focus their attention and effort on the most productive parts of the inevitably large results sets in some forms of search.
- Fifth, multiple forms of query and document sorting will be available: for example pile sorting metaphors as well as simple keyword and Boolean search.
- Sixth, there will be support for collaborative working both groups working in a single location (via for example complex interactive visualisations) and at a distance with advanced support for interaction through video, whiteboards, and multiply-viewed screens, all of which are available now within applications, but NOT integrated and adapted to the patent search task.

We will touch on all these issues in the rest of chapter, going into more depth for some and greater depth in others, largely driven by the clarity we feel can be brought to the issues at the time of writing. The six issues interrelate to each other in a rather complex way, so we will use them as overarching themes, rather than as a rigid framework with which to structure the discussion.

## 20.3 Current State of the Art

In 1999 Larkey [1] produced a paper on what was seen at the time as being a state of the art in patent search technology, and since then many efforts have been made to help patent practitioners making use of patent data to perform their job. They have moved from a digitalised set of patent files to suites of toolboxes allowing them to mine into the data and find information. Only ten years ago, it was all about searching. Nowadays, it is about finding, and finding (ideally) only the relevant documents, which would allow professionals to analyse content and take final decisions. Many commercial providers [2] have gathered and organised subsets of patent collection



and offered computerised access to large companies and administrations. In particular, software suites available from Thompson (i.e. Derwent, Delphion, Micropatent), Questel Orbit, Lexis Nexis, Univentio or Patent Café are popular in the community of patent searchers. The whole list of software producers specialised in patents is available from the PIUG.<sup>1</sup> Dou [3] and Hunt, Nguyen and Rodgers [4] provided an exhaustive analysis of their use. Nevertheless, as explained in Simmons' article [5] the value of data is to be correlated to its quality, which is itself correlated to the power of the computer (-tools) to exploit them.

As well as commercial offerings, there are also many advanced search engines, which have been developed and offered at the disposal of the general public and companies, including:

- EPO Esp@cenet
- USPTO Patent Full-Text and Full-Page Image Databases
- Intellectual Property digital library of WIPO
- SIPO Patent Search and Translation
- KIPRIS search service
- Google Patent Search

For most patent searchers a mixture of high-quality data sources from the commercial publishing sector and free-of-charge data generally mounted under relatively unsophisticated search engines represents the current state of the art in terms of day-to-day practice.

An important distinction needs to be made between invalidity (or validity) searches, and topic or subject-matter searches, like State-of-the-Art or Freedom-to-Operate (see Chap. 1 of this volume).

Invalidity search describes a search triggered by a particular patent application or granted patent against all prior art in a field. It is not limited to a traditional database retrieval exercise but extends to all form of documentation or disclosure potentially invalidating the claims of the patent application (patentability, novelty factors). Subject-matter searches concern searching of a topic in a particular document collection, and frequently start with the patent literature although they may involve the use of non-patent literature at a later stage.

The two kinds of searches exhibit slightly different characteristics. For instance, Fujita et al. [6] have proven that the length of patent document affects relevance in invalidity searches (verbosity hypothesis), whereas it does not in topic searches (scope hypothesis). This work has shown that the verbosity factor (the length) provides a stronger protection for the patent in the view of rights claimed, and is more likely to be relevant to a particular search, since rights claimed are likely to have a broader scope. If documents are considered as being relevant to a topic, then the issue is to demonstrate that they are similar to each other (similar characteristics). This refers to the "cluster hypothesis" described by van Rijsbergen [7]. In a nutshell, a set of documents being relevant to a query should display some kind of similarity. Current retrieval strategies are based on that assumption, which can be tested [8].

---

<sup>1</sup>See: <http://www.piug.org/vendors.php>.

Practically, the cluster hypothesis (at least in its simpler forms) is never fully the case and previous experiments have shown that, for example, cluster-based searches were not effective. One reason may be the vocabulary mismatch problem [9]. The consequence of a weak cluster hypothesis is the possible endangering the relevance feedback mechanisms [10] as in that case, relevant documents might be more similar to non-relevant documents. Remember, relevance feedback assumes that relevant documents have in common some terms (which are not included in the query), and that these terms distinguish the relevant documents from the rest of the collection. Similar should mean relevant, if the cluster hypothesis is correct.

Earlier studies have revealed some other facts about the patent space [11–13]. Because the set of documents exposed to a patent searcher is usually too large to be analysed on an individual basis, professionals have developed strategies to both extrapolate a meta-understanding of the underlying data (through multiple successive queries for example) and reduce logically the size of the corpus under investigation. Experience shows that practical risk minimisation exercises do work, in the case of database exploration/walking, if focussed into a precisely defined technical domain. Patent engineers and examiners, working on a daily basis over large set of documents learn some “meta-information” about the collection, which may be translated into searching strategies. They can then use their meta-information to make better informed judgements about the value of continuing (or resuming) their search at a particular point, as opposed to turning their attention to a more promising avenue of search. Searching similar documents in a corpus is an iterative process, where relevant documents are a priori accessible, thus enabling patent searchers to converge towards their target. Azzopardi [14] has observed that accessibility to the whole collection of data through a search engine is by no means guaranteed by current technology. He demonstrated that a residual set of data is consistently not “visible” to searchers, independently from the query language and the system. That means yet another hypothesis to be tested in the case of patents, where such a bias could be challenging. Patent searchers are even more concerned with this problem of findability than general searchers, because of their regular use of several databases concurrently. Nevertheless, the overall Boolean search process is guided by a set of logical operators allowing them to apply some sort of intuitive reasoning coupled to a firm knowledge of the querying language. A new paradigm would involve not only new searching tools, but new relations towards the machine’s output as well.

Making the machine attempt reasoning instead of the user is a revolution to come. What are required are mechanisms to allow the patent searcher to make judgements about the reliability of the search system: we call this *searcher trust*. In the end, searcher trust in a system is all about being able to analyse, understand and confront output results in a logical systematic way. Achieving such trust would require explanations of the internal processing mechanism leading the displayed results to be accessible and comprehensible to the searcher.

Further as reflected in the previous chapters of this book, patent search has become an active area of research in recent years. More broadly analysis and processing of patents has become a research field per se, and has been recognised as such

by major stakeholders: the community of users [15], the patent offices performing internal research [16], the software developers and the academics [17]. As early as 2000, the ACM SIGIR conference addressed the subject in a special workshop on patent retrieval.<sup>2</sup>

Subsequently (as reported elsewhere in this book) international research campaigns have taken place every few years to benchmark the effectiveness of the latest prototypes against agreed quantitative measures [18]. The NTCIR [19, 20] evaluation campaigns in Japan, as an example, was the first to clearly address the issue of patent analysis and search as a research challenge. Other evaluation campaigns focussed on patent-related issues have been run as part of TREC<sup>3</sup> or CLEF.<sup>4</sup>

These activities have led to the public availability of (admittedly limited) sets of patent data, standardised queries, and assessments of the relevance of retrieved documents.

Stimulated by the Information Retrieval Facility (IRF), since 2008, the ACM Conference on Information and Knowledge Management has included a series of workshops on Patent Information Retrieval (PaIR).<sup>5</sup> These workshops have allowed academic searchers to exchange patent-related research results outside the context of formal evaluations, and have also improved the knowledge of scientific work amongst patent professionals.

The lack of such standardised test collections was probably a major reason for the gap between the SIGIR 2000 workshop and the first PAIR workshop in 2008. Other, more cultural reasons include low levels of European and North American participation in the Japanese-led NTCIR activity, lack of awareness of the economic importance of patent search amongst government research funders, and structural issues, like the USPTO being unable to fund relevant research directly from its budget.

## 20.4 Reflecting Different Sorts of Patent Search

In the first chapter of this book Alberts and colleagues laid out a classification of patent search tasks. Although there are many ways of classifying these tasks, for a technical information retrieval point of view, there are two main dimensions, which can usefully be followed.

First is the range of information to be covered by the search: essentially all information available; all information proven to be publicly available prior to a given date for a patentability search, or limited to enforceable patents and patent applications for a given jurisdiction and date when conducting freedom-to-operate search.

---

<sup>2</sup>See: [http://www.sigir.org/forum/S2000/Patent\\_report.pdf](http://www.sigir.org/forum/S2000/Patent_report.pdf).

<sup>3</sup>See: <http://trec.nist.gov>.

<sup>4</sup>See: <http://clef.iei.pi.cnr.it>.

<sup>5</sup>See: <http://pair.ir-facility.org/>.

Second is the scope of documents, which need to be retrieved. Patent search is often characterised as a high recall search task, but practical experience of working with patent search professionals on formal evaluations of search system effectiveness (see the chapters on NB, TREC CHEM, and CLEF IP elsewhere in this volume) indicate this is not strictly the case. What searchers really need are the most relevant documents up to some limit, which depends on the exact task at hand (time available, type of search, audience for the report etc.), confidence that these really are the most relevant documents; but if fewer than the set limit can be found confidence that highly relevant documents have not been missed.

In current patent search professional practice this is achieved by using multiple search systems with different interfaces and different document collections, with the inevitable cognitive load on the searchers and potential for error switching between systems this entails. One of the few tools, which attempts a more integrated approach is the in-house EPOQUE suite used by the European Patent Office [21].

An ideal future search system will have a single integrated interface which can access multiple collections in a uniform manner, allowing the searcher to specify the numbers of documents they wish to review in detail, and to engage in a simple but (sufficiently) reliable dialogue to give them confidence that they are reviewing the best available documents.

## 20.5 Domain-Specific Intelligence

The current and efficient way of representing knowledge is to distinguish between the description of content elements and their instantiation in terms of references to concrete objects. Those concrete objects could be patent material such as the documents themselves or a sub-part of a patent file (such as the abstract or the claims). The description of content elements is then captured by the so-called ontologies. Ontologies of different levels of abstraction and different types can be used (as described in Wanner et al. [22]):

- A common sense knowledge (core) ontology
- A number of domain-specific ontologies
- A number of patent material specific ontologies that mediate between the highly abstract core ontology and the concrete patent ontologies
- The linguistic ontologies

Such a system can be built up upon an ontology architecture such as the one developed by the IEEE Standard Upper Ontology Working Group (SUMO). A series of ontologies can be defined on the basis of the specific features of a patent document:

- The figures' ontology
- The patent document structure ontology
- The patent bibliography ontology (metadata for the associated to the description of the invention: inventor, date of filing, date of publication, IPC class, etc.)

Of course, technical field specific ontologies have to be added to the search system to allow specialists storing and retrieving specific knowledge (such as in Markush formulae in chemistry or components in electronics) to improve the effectiveness of searches in these fields. Those ontologies can be supplemented by linguistic data extracted from professional thesauri available in the concerned field.

Considerable success has been had recently in using a variety of data driven techniques like Maximum Entropy Markov Models [23] and especially Conditional Random Fields [24, 25] to handle chemical names in building and maintaining such thesauri and ontologies and it therefore seems likely these techniques will be extended to other fields over the next ten years or even less.

Automatic and semi-automatic building and maintenance of ontologies and thesauri is a pre-requisite for the development and adoption of genuinely semantic search systems<sup>6</sup> which are starting to prove they may be effective in contrast to Boolean or more statistical indexing and retrieval systems [26, 27]. However such semantic systems are likely to prove of most value initially in domains where quantities of available technical text (including patents) are small and there are large quantities of available formally codified information about the domain.

In summary then, over the next ten years we are likely to see the adoption of various sorts of technology which augment existing general text search with formally represented information about the nature of the patent search task, the structure of the patent documents themselves, and topical content or domain of the patents or other technical documents being searched.

## 20.6 Multiple Data Sources

As noted above the ideal patent search system would provide a single environment where many sources of data could be searched in a uniform manner. In particular, many forms of patent search require access to the approved patents and pending patent applications from many different patent offices, the academic literature, and ideally any form of public information with a verifiable date.

Of course, this goes way beyond the scope of searches generally conducted at the present time on a day-to-day basis by any patent searcher round the world. In many cases, documents cited in procedures are published in the same country as the case being searched. This does not necessarily reflect a geographical bias in the retrieval, but is commonly due to the fact that the examiners at patent offices prefer to deal with documents in familiar languages and so will often cite a local family

---

<sup>6</sup>It is unfortunate that in the patent search community the term “semantic search” has come to mean two quite different things: on the one hand techniques which rely on opaque semantics emergent from the data like Latent Semantic Analysis [28], Random Indexing [29] and various related techniques which are now quite widely used in patent search and on the other hand techniques which use additional, often completely or partially or completely hand crafted, resources reflecting human understanding of the texts or domains under consideration [30]. Here we mean the latter.

member when available [31]. However, this also underlines the fact that access to the detailed meaning of documents written in foreign languages is still difficult.

On the other hand, facilitating good practice has to be a good thing. The barriers to improvement are now more legal and commercial than technical. Few commercial providers wish to see their existing and new collections available outside their pay walls. However a countervailing force is the Open Science movement and the pressure arising from the US National Institute for Health and others to provide free at the point of use access to at least scientific literature.

It would be possible to write a book on this topic alone, but let us confine ourselves to a small number of points concerning actions needed to improve the situations:

1. There need to be pay mechanisms and models developed which allow the owners and originators of information (including existing publishers) to derive fair rewards from their activities;
2. There needs to be activity to provide improved standardisation of document searching to facilitate automatic indexing and analysis;
3. The search systems and document stores need to be separated to save searchers from having to master new software and interfaces when accessing new sources of information.

Discussions between patent offices and the existing commercial providers have hitherto focussed mainly on the information content of so-called ‘value-added’ data sources and any terms for making them available, rather than developments in retrieval technology [32]. The US-based Coalition for Patent and Trademark Information Dissemination specifically noted that development of new software was not seen as part of the role of the public sector [33].

## 20.7 Multi-linguality

In the early part of this chapter we noted multi-lingualism as a required property of future patent search systems. This is because patent search, of whatever sort, is primarily concerned with the underlying concept of an invention, rather than the language in which it is described. Therefore the patent searcher conducting an invalidity search (for example), wishes to determine whether the idea in a patent has been described in *any language*, in a patent filed at *any patent office* or indeed in an academic paper *in any language* (or indeed any other public information), provided of course the document predates the patent whose invalidity we are seeking to show.

Since much patent litigation covers the precise boundaries of the coverage of a patent the different ways the patent (especially the claims) are expressed in different languages is clearly critical; in other words, so-called equivalent family members may not be a strict word-for-word translation, but differ according to how each national office has granted their version of the application.

Patents then provide a very distinctive sort of challenge in multi-lingual document processing (including machine translation). The need has been clearly established by the wide-spread use of rough-draft statistical translation tools like Google

Translate, despite the fact that these systems do not use models of the domain of invention, patent specific document structure, nor are they integrated in the complex workflows of a patent search office.

As well a research challenge, the patent area provides a challenge and an opportunity for the application of a number of advanced computing technologies. In particular the fact that many patents exist in families with members in different languages, and often with manual translations of at least abstracts in several languages, means that they provide a useful resource for machine learning of various sorts. In particular they can allow the acquisition of statistical translation models specific to the technical vocabulary of a domain (although often the data are rather sparse) and they allow acquisition of the technical vocabulary, potentially with other domain models and language resources like terminologies and ontologies.

Turning to one specific aspect of patent process, there is a considerable body of existing work on multi-lingual patent classification. Even patents describe some solutions of this problem [34]. A number of patent offices and other organisations have investigated and implemented systems of automated categorisation and classification of patent documents using natural language processing and analysis [35]. For instance, WIPO has also developed an online categorisation assistance tool for the International Patent Classification (IPC) system.<sup>7</sup> It is mainly designed to help classifying patents at IPC subclass level, but it also allows the retrieval of similar documents from its database of patent applications [36, 37]. Since that work, many tentative efforts have taken place in order to allow programs categorising automatically patents for limiting this labour-intensive task [38]. Li [39] adopted, for example, a kernel-based approach and design kernel functions to capture content information and various citation-related information in patents. Kim and Choi [40] proposed a  $k$ -NN ( $k$ -Nearest Neighbour) approach for classifying Japanese patent documents automatically, focussing on their characteristics: claims, purposes, effects, embodiments of the invention, instead of only considering the whole full text document. Such an experiment could achieve a 74% improvement of categorisation performance over a baseline system that does not use the structural information of patents. Trappey et al. [41] took another approach and start the classification process by extracting key phrases from the document. This first step is performed by means of automatic text processing to determine the significance of key phrases according to their frequency in text. Correlation analysis is applied to compute the similarities between key phrases, to restrict the number of independent key phrases in the classifier.

It has been shown that machine learning can help in classifying if appropriate data are available for training. Bel et al. [42] have studied two different cases:

1. Bilingual documents are available for training and testing, so that a classifier can learn in two languages simultaneously
2. The classifier learns from language A and then translates the most important terms from language B into A to categorise documents written in language B

---

<sup>7</sup>See: <http://www.wipo.int/ipccat/ipc.html>.

This study based on a Winnow learning algorithm and Rocchio classifier has been applied on a Spanish-English collection and the study has proven that the combination of technique is successful for intrinsic multi-lingual corpora. Many other experiments took place with other languages such as in Czech-English [43] or even other algorithms. For instance, Rigutini et al. [44] employed MT to translate English documents into Italian and classified them by using a Naïve-Bayes approach. Definitely, language is not a barrier for machine learning, just another obstacle for which MT techniques are already effective.

The critical point here is that the feasibility of various forms of advanced multi-lingual patent processing has been demonstrated in research prototypes. These prototypes represent solutions which show that various sections of the patent community are willing to accept less than 100% effective solutions: therefore we are likely to see the adoption of various of these (generally machine-learning based) technologies in patent systems released for general use over the next few years. Their quality will steadily improve over the next ten years, not least because steadily increasing globalisation of the patent system, including enforcement. Globalisation of enforcement will promote multi-lingual invalidity searching which will mean a steady increase of searching in the non-English (and perhaps non-Chinese) patent bases, although English may well remain a dominant language perhaps joined by Chinese in the future.

Of course, increased use and effectiveness of automatic and semi-automatic multi-lingual tools can never supplant the use of human translation: especially, for patents domain translations produced by legal or technical experts with relevant expertise and for particular purposes, like litigation.

## 20.8 Visualisation and Co-operation

Work in patents (and in fact other forms of complex technical information, like gene sequencing data) cannot be adequately represented by simple arrangements of text. On the one hand it is too complex and multi-dimensional to allow this. On the other hand the needs of patent searchers are too complex, subtle and variable to allow one-size-fits all standardised solutions even in areas like presentation of ranked lists of results.

It is important to recognise there are essentially two forms of graphical visualisation of data which are needed by patent professionals:

1. Visualisation of content potentially at the individual document level: content like diagrams, engineering drawings, chemical structures, gene sequences and related text
2. Visualisation of the structure of large information spaces and results to allow the searcher to effectively overview the space and navigate to relevant areas

Obviously there are a number of current applications, which allow various sorts of graphical views of patent data spaces. Considering the problem of large information spaces and results sets, currently searchers often use series of pure textual



Boolean searches (with operators such as “and,” “or,” “not,”) to obtain an overview of the space. Boolean searches are advantageous for experienced searchers who have a clear understanding of the query, as well as the limitations of the database. However, Boolean searches can be difficult for the uninitiated and inappropriate to multiple growing databases. On top of that, learning to master a Boolean search is time consuming, basically consisting of trial and error, which, in a current competitive environment, is not scalable. The goal to achieve is more user-friendly systems to help the researcher to obtain information quickly without a learning phase. Developing methods to access, analyse, and report on patent information in a quicker manner is a challenge shared by both patent offices and patent professionals. More and more forums (such as the IRF Symposium or the European Patent Office Patent Information Conference) aim at gathering the needs and offering the chance to software developers to register the large variety of requirements.

A patent processing system should be more active in assisting the searchers in their repetitive tasks. It could provide suggestions, take the initiative of rewriting the search queries and perform new subsequent searches based on its own understanding.

There is also a growing body of other relevant research. For example, in case of unsupervised neural network clustering, Huang et al. [45] have proposed a SOM (Self-Organising Map) dedicated to patent structures. These authors distinguish between explicit structures (subject, abstract, paragraphs) and implicit structures. Implicit structures refer to writing styles such as “comprising” in the claims (composition style) or “as claimed in” (pre-condition style). This structure analysis occurs as a pre-processing before the SOM and allows a higher robustness to language ambiguities, especially in Chinese language. This clustering is not the first implementation of a SOM for patent documents (see for example [46]), nevertheless the application of clustering is targeting much higher expectations. Indeed, the goal is to compare (cluster) patents showing similar claim contents and to help the patent examiner take critical decisions on the acceptability of a patent application. But note that such a technique, even if perfected, could only assist the general patent searcher in those types of search where the claim language is an important aspect of the target e.g. Freedom-To-Operate searching, and would be less helpful in some aspects of patentability searching where disclosures in the body of the specification are equally or more important. By using a clustering method conflicting patents can be detected, clustered and ranked according to the degree of similarity. On top of that, a graphical representation is a natural way of displaying SOMs. Topic maps are well-know derivatives.

In the past, several initiatives took place to develop visualisation techniques [47, 48]. Mapping tools [49] enabling the display of multiples patent records, but no direct interaction with the end-users has been foreseen. Until recently, probably due to a lack of computer resources (and thus interactivity), many attempts failed. Supported by the need in specific industries or services, such as pharmaceutical research or trend analysis, new developments have nevertheless been proposed [50, 51]. In parallel, a lot of effort has been devoted to text mining techniques, whereas there seems to be a shortage of research on the ability of current technologies to cluster patent data in meaningful and consistent ways.

Text mining techniques are designed to extract non-trivial pieces of knowledge (also called patterns). It is expected that a greater synergy between text mining, knowledge discovery and visualisation is going to improve patent processing methods. Fattori et al. [52] found relevant techniques for the purpose of exploring the patent domain. Visual Data Mining (VDM) for patents would help the end-user building a mental model of a particular dataset and allow him to understand the global structure in addition to the local trends in complex sets. It places the end-user in a position of a cognitive agent navigating in a visual information space and building his own internal cognitive map. In a nutshell, the computer runs heavy processing over millions of records whereas, simultaneously, the user models the virtual document space into its own world. It is desirable that the user establishes a connection between his representation and the system by avoiding the unnecessary underlying complexity.

Tools for VTM (Visual Text Mining) have been proposed in the past [53], but not yet in the context of Industrial Property. Only recently, Yang et al. proposed text-mining approaches in conjunction to independent visual tools to find patent documents [54, 55]. Related attempts have been published in the past, including Topic maps,<sup>8</sup> but mostly concentrated on unstructured corpora such as web content. Patent documents are composed of both structural aspects and a free-style content, which makes the clustering followed by the rendering much less challenging because dependent to the variety of users' expectations (ranging from regular listings to discovery experiences). Results can thus be evaluated more systematically on users' criteria.

Harper and Kelly [56], for example, have show a pile sorting metaphor to be effective in a slightly more complex than average information access task. It would certainly be worth exploring the implications of this result for patent search.

From an internal computer representation to a user-friendly rendering, a series of steps should be put in place. One of them relates to space projection. Multi-dimensional spaces have to be represented on a 2D screen in order to be displayed. The visual representation of the space should nevertheless be compatible with the internal cognitive representation of the user. This poses both a projection issue and a user-interface issue. The projection issue finds its practical solutions through many geometrical techniques [57, 58]. The user-interface problem is addressed through an interactive way of handling subsets of databases and requires computing power capable of scaling with the amount of data. It is essential that methods adequately reflect the content-based neighbourhood relations between documents, according to their similarity. Projections have to be accurate in order to allow the end-user to effectively analyse the space.

Another area where there have been significant advances in visualisation has been in the area of gene sequencing, which of course is of great importance in the patent world (see Havukkala [59] for some examples).

In recent years there have been significant advances in the technologies for virtual meetings, going well beyond the very degraded forms of interaction one gets

---

<sup>8</sup>ISO/IEC 13250:1999.

with simple video conferencing systems. See for example Yu and Nakamaru [60] or Nijholt, Zwiers, and Perciva [61].

Therefore in the next ten years we will see the wide-spread adoption of large format screens, highly interactive complex visualisations, with an ability to reorganise the data on the fly according to the current needs of the searcher. These visualisations will actively support co-operative and team working between different professionals whether co-located or working at different locations.

This will be driven a combination of needing to control the costs of patent work, especially in very high value areas like prosecution, more globalised working (at the same time as trying to reduce travel) as well as technical opportunities and reducing cost of technology.

## 20.9 Towards Integrated Search Solutions

Nowadays, patent professionals are used to many software components allowing a quick overview of the retrieved patents. A good example of such tools is given by Pattools,<sup>9</sup> which performs most of the basic requirements:

- A patent navigator, to display the document content
- An independent claim comparison module, to assess differences between documents claims
- A claim-tree generator to visualise dependencies in claims
- A patent-link generator with family trees

Other online sites such as Toolpat<sup>10</sup> allow access to major global patent databases for prosecution purposes, although it might be argued these are more appropriate for direct use by patent attorneys, rather than professional patent searchers. Another good example of an integrated search environment is SurfIP.<sup>11</sup>

Typically, directly addressing patent searchers needs raise real research problems, such as the need to set up meta-search engines capable of performing searches in all (or some specified) separate search systems and document collections in parallel and then merging the results intelligently before presenting the results back to the user. Another issue is the data fusion aspect implementing the “intelligent” merging of results mentioned above.

However, what is really required is a separation of the tools for the indexing, search, access and analysis of the patent and other data (especially academic literature) from the data itself. This will greatly facilitate the rapid adoption in the patent community of new software developed elsewhere. Integrating these tools goes beyond this: but it will allow the patent community to effectively take part in developments like the open Linked Data initiative<sup>12</sup> and the Semantic Web Services

---

<sup>9</sup>See: <http://www.pattools.com/index.html>.

<sup>10</sup>See: <http://www.toolpat.com>.

<sup>11</sup>See: <http://www.surfip.com/>.

<sup>12</sup>See: <http://linkeddata.org/>.

initiative<sup>13</sup> which will provide the basis of integrated software systems in the future, although there is the danger that the specialised needs of patent searchers may be subsumed under the needs of larger communities and thereof not fully addressed.

## 20.10 Report Generation Support

Now all this advanced technological stuff is all very well, but for the foreseeable future most sophisticated patent searches will result in a paper report (or may be an electronic form like PDF which is essentially 2-D paper which can easily be transmitted and viewed electronically).

Therefore it is important, for the practical patent searcher, that the results of all these advanced information access process can be converted into report form.

We are not going to analyse and review how this is done: this brief section is more of a warning and reminder to technology developers and researchers, but see [62] for a longer discussion of some related issues.

## 20.11 Conclusions

In 1982, Salton started his article describing the SMART retrieval system with the following sentence: “The need to construct complex Boolean queries in order to obtain the benefit of the existing retrieval operations constitutes a substantial burden for the users”. Since the pioneering work of Salton [63] and K. Spark-Jones [64] in the early 70’s considerable progress has been made to make the content of text depositories more easily available to users. Nevertheless, progress has been slow and after more than three decades, many professional users are still facing crucial difficulties in extracting valuable information from datasets. Patent professionals are among them. The issue is no longer to secure a valid search in textual information but instead, to find the relevant piece of information (whatever the data type) and to display it into a framework ready for decision-making.

We have pointed out that a series of different sorts of progress in various scientific fields is needed to address the challenge of processing patent data. Both basic IR technologies and advanced linguistic paradigms are proven to be useful for coping with the multi-lingual nature of patents. Moreover, the continuous exponential growth of patent documents available in the world, raise scalability issues far from being solved. Expectations are at the level of a global economy where language data are at the centre of a full dematerialisation of knowledge. The industrial need to refer to a strong intellectual property portfolio push the trend of enhancing computer tools specialised in processing patents documents. Thanks to coordinated research

---

<sup>13</sup>See: <http://www.swsi.org/>.

efforts such as those of the IRF, involving both professionals and the scientific community, users can finally expect consolidated software suite reaching the level they deserve.

As demonstrated in its time by the pioneering EU project PatExpert [22], emerging technologies addressing these challenges are successful in finding practical research solutions. In the context of managing patent digital data, PatExpert has shown that many concurrent professional issues appeared in the field of Industrial Property (IP). The current chapter focusses on some prospective themes, which the IP community will face in the coming years. As such, and since more and more patent-related projects are initiated, making use of industrial property corpora has become a full subject for academic research and applied research. Expected impacts appear at several orthogonal levels: economical, societal, legal and technical. Although it is not possible to dissociate the legal aspects from the technical one on the user side of the project, it is clear that addressing the Information Technology (IT) side of the issues can solve many practical aspects. For instance, the academic field of Information Retrieval (IR), which has a long history in developing algorithms and methods for exploiting the content of large corpora, has shown interest in focussing its activities on IP. It is now facing a series of use cases potentially showing a great economical impact, thanks to the importance of Internet-based solutions. In parallel, the IP community is morphing from a focus on a librarian-style document management setup to online, to on-the-fly, live and interactive methods.

We predict that in ten years this will drive real changes in the patent search business, leading to the wide-spread adoption of tools which support a truly globalised intellectual property market, and therefore supported shared search-independent of language or location.

**Acknowledgements** The authors would like to acknowledge the contribution of our referees, especially Stephen Adams, for many useful suggestions for improving this chapter.

## References

1. Larkey LS (1999) A patent search and classification system. In: Proceedings of DL-99, 4th ACM conference on digital libraries, pp 179–187
2. Stock M, Stock WG (2006) Intellectual property information: A comparative analysis of main information providers. *J Am Soc Inf Sci Technol* 57(13):1794–803
3. Dou H, Leveillé S (2005) Patent analysis for competitive technical intelligence and innovative thinking. *Data Sci J* 4:209–237
4. Hunt D, Nguyen L, Rodgers M (eds) (2007) Patent searching: tools and techniques. Wiley, Hoboken
5. Simmons E (2006) Patent databases and Gresham's law. *World Pat Inf* 28(4):291–293
6. Fujita S (2007) Revisiting document length hypotheses: NTCIR-4 CLIR and patent experiment at Patolis. Working notes of the 4th NTCIR workshop meeting, pp 238–245
7. van Rijsbergen CJ (1979) Information retrieval, 2nd edn. Butterworths, London
8. Voorhees EM (1985) The cluster hypothesis revisited. In: Proceedings of the 1985 ACM SIGIR conference on research and development in information retrieval, pp 188–196
9. Furnas GW, Landauer TK, Gomez LM, Dumais ST (1987) The vocabulary problem in human-system communication. *Commun ACM* 30(11):964–971

10. Manning CD, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge. Chap 9
11. Wicenc B (2008) Searching the patent space. *World Pat Inf* 30(2):153–155
12. Fujii A, Iwayama M, Kando N (2007) Introduction to the special issue on patent processing. *Inf Process Manag* 43(5):1149–1153
13. Fletcher JM (1993) Quality and risk assessment in patent searching and analysis. In: Recent advances in chemical information; proceedings of the 1992 international chemical information conference, 19–21 October 1992, Annecy. Royal Society of Chemistry, Cambridge, pp 147–156
14. Azzopardi L, Vinay V (2007) Accessibility in information retrieval. In: Proceedings ECIR 2008, pp 482–489
15. Arenivar JD, Bachmann CE (2007) Adding value to search results at 3M. *World Pat Inf* 29:8–19
16. Hassler V (2005) Electronic patent information: An overview and research issues. In: Proceedings 2005 symposium on applications and the internet workshops. SAINT2005:378–380
17. Fujii A, Iwayama M, Kando N (2007) Introduction to the special issue on patent processing. *Inf Process Manag* 43(5):1149–1153
18. Egghe L, Rousseau R (1998) A theoretical study of recall and precision using a topological approach to information retrieval. *Inf Process Manag* 34(2–3):191–218
19. Iwayama M (2006) Evaluating patent retrieval in the third NTCIR workshop. *Inf Process Manag* 42:207–221
20. Fujita S (2007) Technology survey and invalidity search: A comparative study of different tasks for Japanese patent document retrieval. *Inf Process Manag* 43(5):1154–1172
21. Nuyts A, Giroud G (2004) The new generation of search engines at the European Patent Office. In: Proceedings of the 2004 international chemical information conference, 17–20 October 2004, Annecy. Informortics Ltd, Malmesbury, pp 47–56
22. Wanner L, Baeza-Yates R, Brugmann S, Codina J, Diallo B, Escorsa E, Giereth M, Kompatsiaris Y, Papadopoulos S, Pianta E, Piella G, Puhlmann I, Rao G, Rotard M, Schoester P, Serafini L, Zervaki V (2008) Towards content-oriented patent document processing. *World Pat Inf* 30(1):21–33. doi:10.1016/j.wpi.2007.03.008
23. Corbett P, Copestake A (2008) Cascaded classifiers for confidence-based chemical named entity recognition. In: BioNLP 2008: Current trends in biomedical natural language processing, pp 54–62
24. Sun B, Mitra P, Giles CL (2008) Mining, indexing, and searching for textual chemical molecule information on the web. In: Proceeding of the 17th international conference on World Wide Web, pp 735–744. doi:10.1145/1367497.1367597
25. Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc 18th international conf on machine learning, pp 282–289
26. Uren V, Sabou M, Motta E, Fernandez M, Lopez V, Lei Y (2010) Reflections on five years of evaluating semantic search systems. *Int J Metadata Semant Ontol* 5(2):87–98
27. Segura NA, Salvador-Sanchez, Garcia-Barriocanal E, Prieto M (2011) An empirical analysis of ontology-based query expansion for learning resource searches using MERLOT and the gene ontology. *Knowl-Based Syst* 24(1):119–133. doi:10.1016/j.knosys.2010.07.012
28. Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R (1990) Indexing by latent semantic analysis. *J Am Soc Inf Sci* 41(6):391–407. doi:10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASII>3.0.CO;2-9
29. Sahlgren M, Karlgren J (2005) Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Nat Lang Eng* 11(3):327–341. doi:10.1017/S1351324905003876
30. Fernandez M, Lopez V, Sabou M, Uren V, Vallet D, Motta E, Castells P (2008) Semantic search meets the Web. In: Proceedings of the 2008 IEEE international conference on semantic computing (ICSC'08), pp 253–260. doi:10.1109/ICSC.2008.52
31. Michel J, Bettels B (2001) Patent citation analysis; a closer look at the basic input data from patent search reports. *Scientometrics* 51(1):185–201

32. Frackenpohl G (2002) PATCOM—the European commercial patent services group. *World Pat Inf* 24(3):225–227
33. Ebersole JL (2003) Patent information dissemination by patent offices: striking the balance. *World Pat Inf* 25(1):5–10
34. Höfer H, Siemens Business Services GmbH (2002) Method of categorizing a document into a document hierarchy. *European Patent Application EP1244027-A1*
35. Smith H (2002) Automation of patent classification. *World Pat Inf* 24(4):269–271
36. Fall CJ, Töröcsvári A, Benzineb K, Karetka G (2003) Automated categorization in the international patent classification. *ACM SIGIR Forum* 37(1):10–25. doi:[10.1145/945546.945547](https://doi.org/10.1145/945546.945547)
37. Fall CJ, Töröcsvári A, Fievet P, Karetka G (2004) Automated categorization of German-language patent documents. *Expert Syst Appl* 26(2):269–277
38. Loh HT, He C, Shen L (2006) Automatic classification of patent documents for TRIZ users. *World Pat Inf* 28(1):6–13
39. Li X, Chen H, Zhang Z, Li J (2007) Automatic patent classification using citation network information: an experimental study in nanotechnology. In: *Proceedings of the 7th ACM/IEEE joint conference on digital libraries JCDL'07*, pp 419–427. doi:[10.1145/1255175.1255262](https://doi.org/10.1145/1255175.1255262)
40. Kim JH, Choi KS (2007) Patent document categorization based on semantic structural information. *Inf Process Manag* 43(5):1200–1215
41. Trappey AJC, Hsu FC, Trappey CV, Lin CI (2006) Development of a patent document classification and search platform using a back-propagation network. *Expert Syst Appl* 31(4):755–765
42. Bel N, Koster CHA, Villegas M (2003) Cross-lingual text categorisation. In: *Proceedings ECDL. LNCS*, vol 2769, pp 126–139
43. Olsson JS, Oard DW, Hajic J (2005) Cross-language text classification. In: *Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval*, pp 645–646
44. Rigutini L, Maggini M, Liu B (2005) An EM based training algorithm for cross-language text categorization. In: *Proceedings of the IEEE/WIC/ACM international conference on web intelligence*, pp 19–22
45. Huang SH, Ke HR, Yang WP (2008) Structure clustering for Chinese patent documents. *Expert Syst Appl* 34(4):2290–2297
46. Ma Q, Nakao K, Enomoto K (2005) Single language information retrieval at NTCIR-5. In: *Proceedings of NTCIR-5 workshop meeting*, December 6–9, 2005, Tokyo, Japan
47. Suh JH, Park SC (2006) A new visualization method for patent map: Application to ubiquitous computing technology. *LNAI*, vol 4093, pp 566–573
48. Fischer G, Lalyre N (2006) Analysis and visualisation with host-based software—The features of STN®AnaVist™. *World Pat Inf* 28(4):312–318
49. Blanchard A (2007) Understanding and customizing stopword lists for enhanced patent mapping. *World Pat Inf* 29(4):308–316
50. Eldridge J (2006) Data visualisation tools—a perspective from the pharmaceutical industry. *World Pat Inf* 28(1):43–49
51. Kim YG, Suh JH, Park SC (2008) Visualization of patent analysis for emerging technology. *Expert Syst Appl* 34(3):1804–1812
52. Fattori M, Pedrazzi G, Turra R (2003) Text mining applied to patent mapping: a practical business case. *World Pat Inf* 25(4):335–342
53. Lopes AA, Pinho R, Paulovich FV, Minghim R (2007) Visual text mining using association rules. *Comput Graph* 31(3):316–326
54. Yang Y, Akers L, Klose T, Yang CB (2008) Text mining and visualization tools—Impressions of emerging capabilities. *World Pat Inf* 30(4):280–93
55. Yang YY, Akers L, Yang CB, Klose T, Pavlek S (2010) Enhancing patent landscape analysis with visualization output. *World Pat Inf* 32(3):203–220. doi:[10.1016/j.wpi.2009.12.006](https://doi.org/10.1016/j.wpi.2009.12.006)
56. Harper DJ, Kelly D (2006) Contextual relevance feedback. In: *Proceedings of the 1st international conference on information interaction in context*, Copenhagen, Denmark, October 18–20, 2006. *IiX*, vol 176. ACM, New York, pp 129–137. doi:[10.1145/1164820.1164847](https://doi.org/10.1145/1164820.1164847)

57. Paulovich FV, Minghim R (2006) Text map explorer: a tool to create and explore document maps. In: Information visualization (IV06). IEEE Computer Society Press, London
58. Paulovich FV, Nomato LG, Minghim R, Levkowitz H (2006) Visual mapping of text collections through a fast high precision projection technique. In: Information visualization (IV06). IEEE Computer Society Press, London
59. Havukkala I (2010) Biodata mining and visualization: novel approaches. World Science Publishing Co, Singapore
60. Yu Z, Nakamura Y (2010) Smart meeting systems: A survey of state-of-the-art and open issues. *ACM Comput Surv* 42(2):1–20. doi:[10.1145/1667062.1667065](https://doi.org/10.1145/1667062.1667065)
61. Nijholt A, Zwiers J, Peciva J (2009) Mixed reality participants in smart meeting rooms and smart home environments. *Pers Ubiquitous Comput* 13(1):85–94. doi:[10.1007/s00779-007-0168-x](https://doi.org/10.1007/s00779-007-0168-x)
62. Adams S (2005) Electronic non-text material in patent applications—some questions for patent offices, applicants and searchers. *World Pat Inf* 27(2):99–103
63. Salton G (1971) The SMART retrieval system—experiments in automatic document processing. Prentice-Hall, Inc, Upper Saddle River
64. Sparck Jones K (1972) A statistical interpretation of term specificity and its application in retrieval. *J Doc* 28(1):11–21



# Index

## A

Abbreviation, 289  
Aboutness, 264, 266, 278, 279, 284  
Accessibility, *see* retrievability, 393  
Accuracy, 239, 254, 257  
ACM Conference on Information and Knowledge Management, 394  
ACM SIGIR, 394  
Acronym, 289  
Agglutination, 249  
Analytic accountability, 231  
Analytic cycle, 224  
Anaphora resolution, 334, 341  
Anchoring, 136  
Annotate, 207  
Applicant, 22  
Artificial Intelligence, 239  
Asian languages, 249  
Assessor, 118, 178  
Assignee, 22  
Automated patent classification, 240  
Automated patent classifier, 250  
Automatic text processing, 398

## B

Backward citation searching, 22  
Basic Local Alignment Search Tool (BLAST), 36  
Bayesian algorithms, 251  
Bayesian classification, 294  
Beilstein, 31  
Bibliographic data, 16, 22, 221, 223  
Binary preference, 120  
BioCreative, 110  
Biomedical, 112, 334, 340, 360  
Biosequence, 34  
BiTeM Group, 294

Bm25, 155, 158, 187  
Boolean logic, 15, 30, 46, 181  
Boolean operators, 21, 62, 150, 153, 227, 228, 230, 312, 366  
Boolean search, 227, 391, 393, 400  
Bootstrap problem, 250

## C

CAS, 295, 344, 346, 351, 352  
Categorization, 203, 263, 266  
Chemical Abstracts Service (CAS), 25  
Chemical graph, 346  
Chemical patent, 343  
Chemical reaction search, 110  
Chemical structure diagram, 344  
Chemical Structure Searching, 28  
Chemical structures, 344, 348, 349, 351, 353, 354  
Chemical Substance Registries, 25  
Chemical synonyms, 123  
Chemistry-specific methods, 123  
Cheminformatics, 343, 345–347, 354  
Citation, 91–93, 118, 334, 377, 380  
Citation analysis, 40, 73  
Citation graph, 221  
Citation search, 22  
Claim  
    dependent, 198, 199, 201, 208  
    independent, 198, 201  
    merging, 209, 213  
Claim-specific language, 197  
Claims section, 19  
Classification, 16, 23, 39, 60, 89, 91, 93, 226, 240, 263, 264, 269, 279, 350  
Classification code hierarchy, 290, 296, 302  
Classification code structure, 289  
Classification consistency, 242

- Classification engine, 271
  - Classification hierarchy, 243, 288, 294
  - Classification search, 289
  - Classification symbol, 291
  - Classification system, 288
  - Classifier, 96, 246
  - Clearance search, 288
  - CLEF, 50, 70, 87, 132
  - CLEF IP, 294, 395
  - Closed Substructure Search (CSS), 30
  - Cluster hypothesis, 392
  - Clustering, 60, 220, 221, 233, 335, 401
  - Co-occurrence, 383
  - Coalition for Patent and Trademark Information Dissemination, 397
  - Collaborative visual analysis, 231
  - Collision, 347
  - Collocations, 258
  - Competitive intelligence, 127
  - Composition style, 400
  - Compound, 26, 28, 31, 55, 111, 115, 249, 258, 292, 344, 350
  - Conceptual graph, 200
  - Conditional Random Fields, 396
  - Confidence interval, 174
  - Connection table, 344–348, 351
  - Consistency, 363
  - Content model, 185
  - Content words, 266
  - Controlled Vocabulary, 24
  - Convexity, 156
  - Corpora, 269
    - CLEF-IP, 270
  - Corpus, 72, 90, 111, 168, 323, 362
  - Cost, 9
  - Cranfield, 131, 133
  - Cross-lingual, 87
  - Cross-validation, 272
  - Curated databases, 124
- D**
- DARC system, 351, 353
  - Data preprocessing, 202
  - Data set, 201, 380, 384
  - Data structure, 204
  - Databases, 182
  - Datalog, 182
  - DBpedia, 362
  - DCG, *see* discounted cumulative gain
  - Decision rules, 251
  - Decomposition algorithm, 211
  - Delphion, 392
  - Dependency
    - graph, 266, 267
    - model, 263, 267
    - triple, 263, 266, 267, 285
  - Depth of classification, 292
  - Derwent, 392
    - Chemical Code, 352
    - Chemistry Resource (DCR), 26
    - fragmentation codes, 30
    - World Patents Index (DWPI), 22, 26
  - Design patent, 288
  - Diagram, 246
  - Dialog, 8
  - DIR, *see* distributed IR
  - Discounted cumulative gain, 81
    - normalized, 82, 97
  - Distributed IR, 182
  - Distributed processing, 308
  - Distribution imbalance, 247
  - DNA Data Bank of Japan, 37
  - Document representation, 284
  - Drug-discovery, 349
  - Due diligence search, 14
  - DuPont/IFI code, 352
- E**
- E-discovery, 167
  - ECLA, 16, 240, 244, 289, 290, 293
  - Effectiveness, 48, 70, 74, 83, 86, 97, 125, 147, 167, 280, 294, 373, 399
  - Efficiency, 8, 60, 97, 119, 182, 259, 287
  - Electronic discovery, *see* e-discovery
  - Elsevier, 8, 338
  - Embase, 31
  - Engineering Drawings, 6, 38, 399
  - Entity recognition, 122
  - Enet, 392
  - EPOQUE suite, 395
  - European Bioinformatics Institute, 37
  - Evaluation, 47, 50, 69–71, 83, 92, 201, 210, 212, 336, 359, 381
    - system-based, 70
    - user-based, 70
    - user-centred, 49
  - Evaluation metric, 140
  - Evidence of Use, 10
  - Exact Search (EXA), 30
  - Experiment, 67, 71, 97, 152, 159, 264, 270, 272, 284, 398
- F**
- F-measure, 78, 253
  - F1 measure, 169, 252
  - Failure analysis, 131, 177
  - Family Search (FAM), 30, 36

FASTA, 36  
Feature constraints, 314  
Feature vector, 335  
Fee-based sources, 8  
FI/F-Term, 16  
Filtering, 220, 223, 226, 230  
Findability, 150, 393  
Fingerprint, 348, 349  
FIRE, 50, 70  
FlyBase, 330, 336  
Foreign Spelling, 289  
Formal textual query languages, 227  
Format uniformity, 7  
Forward citation searching, 22  
Fragment descriptor, 352  
Fragment-based screening stage, 352  
Free sources, 8  
Freedom-to-operate, 12, 126, 128, 242, 392, 400  
Frequency variation, 350  
Full-text search, 16, 20  
Fuzzy FRELS, 353

## G

GATE, 202, 204, 205, 319, 321  
GBJ2 Biosequence Search Strategy, 37  
Generic structures, 343, 349, 354  
GenomeQuest's GenePAST/Fragment search, 37  
Geo-spatial data, 224  
Google Patent Search, 392  
Grammatical relation, 334, 335, 339  
Grammatical structures, 199  
Graph isomorphism, 345, 347  
Graph view, 226  
Graph-based generic system, 352  
Graphic information, 246  
GREMAS code, 352

## H

Hardware, 195, 220, 231, 322  
Hashing, 335, 347  
Hierarchical data, 204, 224  
Hierarchical parser, 228  
Hierarchical query, 228  
High recall, 16, 21, 27, 129, 141, 145, 395  
High risk, 129, 141  
Hill system, 25  
Homology search, 35  
Homology variation, 350, 351  
Human annotators, 320  
Human assessor, 72  
    agreement, 74  
Human classification, 241

Human expert, 361  
Human-computer interaction, 220  
Hyponym, 378  
Hyponym, 378

## I

IEEE Standard Upper Ontology Working Group, 395  
IFI CLAIMS Compound Vocabulary, 26  
IFICDB, 31  
Image processing, 335  
Image search, 329  
Image similarity search, 227, 230  
Index, 54, 154, 310  
Indri, 96, 121, 122, 296, 297  
INEX, 70, 133  
Inflection, 249  
Information extraction, 62, 197, 217, 308, 319, 330  
Information management, 6  
Information need, 72  
Information Types, 6  
Information Visualization, 218, 219  
Infringement search, 242  
Integrated search solutions, 402  
Intellectual Property digital library of WIPO, 392  
Interactive visualisation, 217, 218, 220, 231, 233  
Interface, 395  
International Chemical Identifier, 345  
Interpolated precision, 75  
Invalidity search, 375, 392  
Inventor, 23  
IPC, 16, 90, 91, 198, 240, 243, 290, 293, 316, 358, 381, 398  
IPC Advanced Level, 243  
IPC classes, 111, 113  
IPC Core Level, 243  
IPCCAT, 254  
IR evaluation, 47, 50, 92, 109, 125, 131, 134, 139  
IR model, 58  
    best-match, 156  
    boolean, 58  
    exact match, 149  
    hybrid, 150  
    logic-based, 181  
    ranked, *see also* best-match  
Iteration, 40, 231  
Iterative query improvement, 226  
IUPAC, 344

**J**

JAPE, 204, 208

**K**

K-Nearest Neighbour, 239, 251, 256, 294, 398

Keyword search, 12, 26, 31, 40, 227, 289, 316, 335

KIPRIS search service, 392

**L**

Laboratory test, 49, 137

Laboratory-style evaluation, 131, 138

Language Model, 187, 266

Large-Scale Logical Retrieval, 193

Learning algorithm, 121, 399

Legal status, 13, 24, 242

Lemur, 96, 121, 122

Lexis Nexis, 392

Linear notation, 344–346

Linguistic, 249, 294, 319

patterns, 208, 331

processing, 249

techniques, 263, 264

Linked Data, 358, 402

Linking, 224, 232

List-based interfaces, 223

LM, *see* Language Model

Logical tree, 350

LSA, 60

LSLR, *see* Large-Scale Logical Retrieval

Lucene, 122, 310, 335, 338, 365

**M**

Machine learning, 220, 233, 399

Machine translation, 21, 97, 397

MAP, *see* mean average precision

MAREC, 90, 92, 159, 201, 316

Markush, 344

claim, 29

DARC system, 31

formula, 396

search, 28

structure, 30, 349, 351, 353

Marpat, 31, 351

Master Classification Database (MCD), 242

Matching technique, 135, 348

Maximum Entropy Markov Models, 396

Mean average precision, 52, 80, 97, 120, 298, 336

Mean reciprocal rank, 120, 373, 388

Measure, 48, 50, 52, 69, 74, 83, 97, 381

estimation, 170, 178

Medlars search service, 131

Medline, 31

Merged Markush System (MMS), 31

MeSH, 325, 336, 341

Meta-search engines, 402

Metadata, 6, 22, 46, 55, 97, 309, 320, 358, 369  
search, 227

Micropatent, 392

Molecular representation, 345

Morgan algorithm, 346, 347

MRR, *see* mean reciprocal rank

Multi search back-end, 229

Multi-lingual patent processing, 399

Multilingual, 41, 88, 97, 397

Multilingual patent classification, 398

Multinomial logit, 294

Multiple coordinated views, 224

Multipunch code, 26

**N**

*n*-gram processing, 259

*n*-gram rule, 259

Naïve-Bayes approach, 399

Named entity, 335

recognition, 62, 334, 341

National Center of Biotechnology Information,  
37

Natural language, 219, 248, 316, 320

processing, 15, 62, 197, 200, 217, 264, 318,  
334

Neural network, 247, 251

NIST: National Institute of Standards and  
Technology, 70, 294

NLP, *see also* Natural language processing

Normalization, 269

morphological, 269

syntactic, 268

Normalized discounted cumulative gain, 120,  
298

Novelty, 242

Novelty search, 288

NP-complete computational problem, 347

NTCIR, 50, 70, 88, 92, 132, 294, 375, 394

**O**

Okapi, *see* bm25

One-to-one mapping, 352

Ontological similarity, 295

Ontology, 62, 258, 358, 360, 361, 365, 367,  
369, 395

Open Science, 397

Operator, 315

AND, 227, 315

IN, 316

OR, 227, 315

- Operator (*cont.*)  
 OVER, 316  
 repetition, 316  
 Optical Character Recognition (OCR), 250
- P**  
 PaIR, 394  
 Panning, 220, 224  
 Parallel, 40, 55, 119, 219, 224, 271, 320, 330  
   architecture, 181, 192  
 Parameterised query, 231  
 Pareto Principle, 247  
 Parser, 197, 200, 210, 211, 269–271, 279, 285, 334  
 Parsing, 264, 266  
 Patent application, 4, 11, 19, 88, 113, 127, 225, 274, 287, 380  
 Patent Café, 392  
 Patent categorization, 294  
 Patent Cooperation Treaty (PCT), 20  
 Patent document, 87, 111, 202, 270, 310, 317, 359, 362, 370  
 Patent family, 13, 23, 32, 37, 89, 221  
 Patent landscape analysis, 12, 126, 391  
 Patent life cycle, 4, 88  
 Patent mining, 376  
 Patent network, 221  
 Patent Portfolio Analysis, 14  
 Patent prosecution, 4, 22, 40, 402  
 Patent Search Types, 9  
 Patent similarity, 294  
 Patent structure, 245, 400  
 Patent term, 4, 13, 373  
 Patentability, 12, 126, 127, 242, 287  
 Patentee, 22  
 Patentese, 19  
 PatExpert, 221, 310, 404  
 Pattern matching, 319  
 Patterns, 204  
   characterized-pattern, 205  
   claim-subject, 205  
   composition-pattern, 206  
   description-pattern, 207  
   nested-sentence-pattern, 206  
 Pattools, 402  
 PatViz, 217, 221  
 PDF, 331, 341  
 Polysemy, 248, 250  
 Pooling, 73, 92, 118, 134, 170  
 Position variation, 350  
 Post search analysis, 9  
 Pre-classification, 241, 254, 255  
 Pre-condition style, 400  
 Precision, 16, 50, 71, 74, 75, 97, 120, 130, 139, 141, 167, 252, 269, 274, 282, 338, 384  
   average precision, 78  
   R-precision, 80  
   rank-biased, 82  
 Predicate, 185, 190, 339  
 Prior art, 12, 91, 113, 118, 222, 242, 257, 288, 377  
 Product measures, 140  
 Proximity, 15, 296  
 PubChem, 122  
 Purdue, 294
- Q**  
 Quality, 9, 41, 52, 132, 149, 202, 212, 217, 245, 258, 294, 359, 390  
 Query, 323  
   analytical, 183  
   annotation, 314, 315  
   boolean, 72, 149, 175, 312, 339  
   complex queries, 312, 329  
   generation, 153  
   keyword, 72  
   language, 312, 338  
 Query expansion, 62, 331, 340, 379  
 Query formulation, 221, 223  
 Query Performance Analyser, 141  
 Query syntax, 227  
 Query types, 313  
 Questel Orbit, 14, 392
- R**  
 RDF, 190, 312, 360, 368  
 Re-classification, 241, 256  
 Reasoning, 62, 136, 180, 183, 190, 266, 312, 393  
 Recall, 50, 71, 74, 75, 97, 120, 129, 139, 141, 167, 252, 269, 274, 282, 298, 338, 384  
   measurement, 147  
 Reduced graph, 352  
 Reference structure, 349  
 Registry, 25, 31  
 Regular expression, 202, 205, 315, 319  
 Relational analysis, 226  
 Relevance feedback, 56, 122, 148, 179, 393  
 Relevance judgement, 49, 72, 83, 92, 118, 178  
 Relevance judgements  
   false positive rate, 178  
   manual, 118  
 Resource description framework, *see* RDF  
 Retrievability, 148, 150  
   distribution of, 151

- Retrieval effectiveness, 141  
Rocchio, 251, 399
- S**
- Sampling, 73, 170, 171  
    stratified, 173  
Scalability, 109, 182, 259, 308, 320, 360  
Scope hypothesis, 392  
Screening search, 348  
Searchability, 6  
Segmentation, 249  
Selection management, 225, 226, 230, 231  
Selection semantics, 225  
Self-Organizing Map (SOM), 400  
Semantic annotation, 308, 310, 316  
    measurements, 318, 320  
    parallel, 321  
Semantic data, 181, 185, 186, 195  
Semantic relatedness, 294  
Semantic search, 45, 61, 227, 230, 306, 396  
Semantic Web, 360  
Semantic Web Services initiative, 403  
Sequence Code Match (SCM), 35  
Sequence information, 35  
Sheffield Generic Structures Project, 352  
Similar Property Principle, 349  
Similarity, 15, 60, 200, 242  
    search, 349, 353  
SIPO Patent Search and Translation, 392  
SMART, 71, 131, 403  
SMILES notation, 345  
Snippet, 54, 311, 338  
Sorting, 15, 26, 46, 57, 157, 223, 226, 391  
Source selection, 193  
Space projection, 401  
SQL, 182  
State-of-the-Art, 10, 126, 127, 392  
Statistic processing, 249  
Statistical phrases, 264  
Statistical test, 80  
Stemming, 55, 154, 160, 259  
STN, 14, 30  
Strategy, 10, 21, 30, 37, 126, 184, 186, 190, 227  
Structural information, 343, 353  
Structural variation, 350–353  
Structure information, 111  
Structure search, 110, 117, 346, 352  
Subgraph isomorphism, 345, 347, 348, 352  
Subject-based databases, 25  
Subject-matter search, 392  
Substituent variation, 350  
Substructure search, 30, 347  
Support Vector Machine, 239, 251  
SureChem database, 353  
SurfIP, 402  
Synonym, 25, 289, 337, 339, 383  
Synsets, 294  
Syntactic parsing, 334  
Syntactic phrases, 264  
Syntactic relations, 266  
Syntax diagram, 227  
System performance, 48, 69, 75, 80, 126, 140  
Systematic nomenclature, 344, 346
- T**
- Tables, 333  
Tag cloud, 226, 229, 230  
Task, 71, 87, 91, 113, 168, 182  
    ad-hoc, 168  
    batch, 168  
    interactive, 168, 178  
Taxonomy, 24, 62, 240, 243, 244, 358–360, 369  
Technology survey, 115, 118, 375  
Temporal data, 224  
Term Distillation, 376  
Term extraction, 365  
Term selection, 96, 281  
Terminological evolution, 250  
Terrier, 96  
Test collection, 50, 69, 71, 83, 87, 110, 131, 148, 172, 263, 375, 394  
Text analysis, 123, 217, 218  
Text mining, 217, 308, 331  
Tf-idf, 61, 155, 157, 187, 191, 375  
The Jackson Laboratory, 37  
Thesaurus, 24, 56, 62, 325, 373, 378f  
Thompson, 392  
Toolpat, 402  
Topic, 72, 74, 91, 96, 112, 117, 133, 168, 247  
Topic variability, 140  
Topological search systems, 30  
Tradename, 31  
Training data, 50, 56, 96, 178, 240, 250, 257, 271, 309, 334, 398  
Translation, 21, 89, 97, 250, 373  
TREC, 50, 70, 87, 131, 158, 394  
    Chemical IR Track, 110, 294, 376, 395  
    CiQA Track, 137  
    Genomics Track, 109  
    Hard Track, 137  
    Legal Track, 167  
    Million Query Track, 176  
    Terabyte Track, 170  
Tree-search procedure, 347  
Triples, 263f, 281, 312, 360

**U**

Univentio, 392  
Unsupervised neural network clustering, 400  
US National Institute for Health NIST, 397  
User confidence, 140  
User effort, 81  
User interface, 9, 49, 57, 69, 220, 313, 331, 367, 401  
User model, 82, 401  
User test, 137  
USGENE, 37  
USPC, 16, 290  
USPTO, 4, 92, 111, 198, 289, 316, 362, 392  
Utility patent, 288

**V**

Validity, 13, 126, 149, 233, 242, 288, 374, 390  
Value-added data sources, 397  
Value-Added Indexing, 24  
Vector-space model, 59, 265  
Verbosity hypothesis, 392  
Visual analytics, 217f  
Visual Data Mining (VDM), 401  
Visual feedback, 218, 233  
Visual front-end, 227  
Visual patent analysis, 218

Visual similarity, 331  
Visual Text Mining (VTM), 401  
Visualising textual information, 219  
Visualization, 369, 399  
Vocabulary, 16, 61, 135, 156, 200, 249, 306, 325, 359, 398  
Vocabulary mismatch, 377, 393

**W**

Weighting, 54, 74, 96, 191, 248, 287, 300  
Wikipedia, 248, 306, 357f  
Wildcard, 54, 336  
WIPO, 111, 239f, 289f, 358, 392  
Wiswesser Line Notation, 345  
Word sense disambiguation, 258  
WordNet, 294  
Workflow, 307, 390

**X**

XML, 70, 90, 111, 160, 182, 245, 295, 316, 331, 366

**Z**

Zipf's law, 47, 248  
Zooming, 220