PaEffExtr: A Method to Extract Effect Statements Automatically from Patents

Na Deng[1](✉), Xu Chen[2], Ou Ruan[1], Chunzhi Wang[1], Zhiwei Ye[1], and Jingbai Tian[1]

[1] School of Computer, Hubei University of Technology, Wuhan, China
iamdengna@163.com
[2] School of Information and Safety Engineering,
Zhongnan University of Economics and Law, Wuhan, China
chenxu@zuel.edu.cn

Abstract. Patents contain a lot of technical, economic and legal information, and they are the main references of enterprises' technological innovation. As a tool of patent analysis and mining, technology/effect matrix provides important support for technological innovation and avoidance. In the process of building technology/effect matrix, most of current technical efficiency annotation is by manually work, which requires heavy labor. Considering the distribution and morphological characteristics of patent abstract texts, this paper proposes a multi-features fused scoring algorithm named PaEffExtr, which automatically extracts effect statements from patent abstract texts. The experimental results show that the algorithm has good recall and accuracy.

1 Introduction

With the development of society, people are more and more aware of the tremendous changes in our life brought about by innovation. As one of the most important ways to protect innovation, patent has been paid more and more attention. More and more patents are accumulated in the worldwide since the amount of patents applications increases year by year. Because patents contain rich technology, economy and law information, patent analysis and mining has become an important research topic in the field of data mining. Nowadays, with the rapid development of market economy, enterprises have to seize the highland of technology for sustainable development. Technology/effect matrix is a tool of patent analysis and mining. It can help enterprises to find technology vacant areas and minefields, and provides important support for technological innovation and avoidance. In the process of building technology/effect matrix, the annotation of technology/effect is a rather important step. At present, technology/effect is mostly by manual annotation, requiring a lot of heavy manual labor. In addition, manual annotation is subjective, for the same patent, different annotators may have different ways, which may bring hidden trouble for patent mining. This paper aims to solve these problems.

L. Barolli and O. Terzo (eds.), *Complex, Intelligent, and Software Intensive Systems*,
Advances in Intelligent Systems and Computing 611, DOI 10.1007/978-3-319-61566-0_62

2 Related Work

In recent years, there are many research on patent analysis and mining at home and abroad. [1] investigated multiple research questions related to patent documents, including patent retrieval, patent classification, and patent visualization. [2] used OPTICS algorithm and k-nearest neighbor to implement clustering analysis of patent information. [3, 4] tried to focus on vacant technology forecasting, by using K-medoids or Bayesian. [5] gave a survey on different text clustering techniques for patent analysis. [6] used self-organizing map (SOM) approach to cluster patents into different quality groups and used support vector machine (SVM) to build up the patent quality classification model. [7] studied the patent document classification problem by deep learning. [8] focused on keyword strategies for applying text-mining to patent data and addressed four factors about key words.

In the domain of patent technology effect matrix, there are also some but not many research [9–15]. Japanese scholars [9, 10] were the earliest to study on technology effect matrix of Japanese and English language patents. [11] applied semantic role labeling to create technology-effect matrix. [12] proposed a method for matrix structure construction based on feature degree and lexical model. [13] gave one kind method based on conditional random field model (CRFs) to recognize effect phrases.

In the authors' previous work about patent analysis and mining [14–17], we mainly focused on the removal of stop words in patents, intelligent recommendation of the traditional Chinese medicine patents and effect annotation. In the research about annotation, we found that the same patent inventor has his/her preferred style of writing; thus, using co-training method, effect statements' extraction is divided into chain extraction and keywords extraction, which iteratively annotate effect statements in patent abstract. However, the limitation of this method is that it is easy to produce misjudgment. That is, some statements that are closely related to each other but actually not effect statements will be deemed as effect statements falsely. In this paper, making use the distribution and morphological characteristics of patent effect statements, and trying to make the extraction algorithm more general, but not limited to a patent inventor, we propose a multi-features fused scoring algorithm for automatic extraction of effect statements.

The rest of paper is organized as follows: Sect. 3 analyzed and summarized the characteristics of Chinese patent abstract, including distribution and morphological characteristics of effect statements. Section 4 described the automatic annotation algorithm PaEffExtr in detail. Section 5 analyzed and explained the experimental results. Section 6 concluded the paper and prospected the future work.

3 Characteristics of Patent Abstracts

Generally, a patent text consists of title, abstract, claim and specification. Patent abstract is a summary of the whole content of the patent text. It is short, but contains the composition structure of the invention, technologies used, design principles, functions, scope of application and other important information. Therefore, patent abstract is the data source of many patent mining experiments. In patent abstract, there is usually a description of the function and application scope of the invention, which is called as patent effect. The purpose of this paper is to automatically extract effect statements from Chinese patent abstracts.

In order to facilitate the following explanation, two definitions are given as follows:

Definition 1: patent effect statements
A collection of statements describing the function and application scope of the invention in the text of patent abstract, denoted as ES. From the perspective of linguistics, the elements in this collection are not necessarily close to each other in the abstract text.

Definition 2: patent effect clause
The element in patent effect statements, denoted as ec. From the perspective of linguistics, patent effect clause may be a single sentence, and also may be a clause in a long sentence.

So we can say that ES = {ec}.

After observing a large number of patent abstracts, we found that effect statements had two obvious characteristics.

(1) Distribution characteristic: in patent abstracts, the positions of effect clauses follow certain rules. In many cases effect clauses appear at the end of the abstract, in a few cases appear in the head of the abstract, in rare cases in the middle of the abstract. Sometimes, all of the effect clauses in a patent abstract are distributed in multiple places, but in many cases, all the effect clauses appear in a continuous way.
(2) Morphological characteristic: because effect statements describe the function and the scope of application of the invention, there are often specific clue words in effect clauses. These clue words may be used to guide the emergence of an effect clause, and may also indicate which aspects have changed, what changes have been made and so on.

According to different situations, we divide the clue words into the following categories.

(1) leading word: a word used to guide the emergence of effect clause. For example: “have”, “can”, “apply to”, “used to”, “make”, etc.
(2) facet word: a word used to indicate which aspects have changed brought by a patent invention. For example: “cost”, “performance”, “quality”, “efficiency”, etc.
(3) changing word: a word reveals what changes have been made by a patent invention. For example: “improve”, “simple”, “lower”, “avoid” and so on.
(4) degree word: a word used to indicate the extent to which a patent invention have changed. For example: “significant”, “obvious”, etc.

4 Multi-features Fused Scoring Algorithm

According to the introduction above, we find that effect clauses in patent abstract have its obvious distribution and morphological characteristics. Those clauses at specific locations and containing clue words are more likely to be effect clauses than other clauses. Therefore, we design a multi-features fused scoring algorithm, based on the location information and whether containing clue words, to give score to each clause, and choose those with high scores as effect clauses.

4.1 Calculation of Distribution Score

There is no mandatory requirement for the writing of abstract text of patents, so patent applicants usually write according to their own habits and preferences. Through the observation we found that in many cases, functions and application scope of patents are located in the tail of the abstracts, in few cases are in the head, in rare cases are in the middle, even sometimes there is not any effect statement in some abstract. In addition, the use of punctuation marks is also very arbitrary. Some applicants are accustomed to use periods to separate the patent structure, technology, design principle, function and application range, some tend to use a semicolon, and some only use commas directly. In this paper, we will use the comma, semicolon, periods etc. as delimiter, to separate abstract into clauses. We calculate distribution score using the following method.

With regard to a patent abstract text T, we use C to represent the set of all its clauses. For the *ith* clause c_i, its distribution score is calculated as:

$$D_i = \begin{cases} \gamma_1 & when\ 0 < i < \frac{N}{3} \\ \gamma_2 & when\ \frac{N}{3} \le i < \frac{2N}{3} \\ \gamma_3 & when\ \frac{2N}{3} \le i \le N \\ & \qquad (\gamma_1 + \gamma_2 + \gamma_3 = 1) \end{cases} \tag{1}$$

In Formula 1, $|C| = N$, D_i represents the distribution score of c_i. The abstract text is divided into 3 parts, each of which is given a single weight.

4.2 Calculation of Morphological Score

Because the clauses containing clue words are more likely to be effect clauses than others, so we check whether the clause contains clue words and how many clue words to calculate morphological score.

The collection of clue words is the key to calculate morphological score. Through the observation, we found that clue words appear frequently in effect statements, so we use statistical methods to find them. By manually annotating effect clauses of a certain number of patents, looking for high-frequency words, artificial screening, and several rounds repeated, we constitute a clue words set called ClueWords.

With regard to a patent abstract text T and *ith* clause, the morphological score is calculated as follows.

$$M_i = \lambda k_i \tag{2}$$

In (2), k_i represents the number of clue words contained in c_i.

4.3 Algorithm PaEffExtr

In summary, for the *ith* clause in a patent abstract text T, the score is calculated as follows:

$$Score_i = \alpha D_i + \beta M_i \qquad (\alpha + \beta = 1) \tag{3}$$

Here, α and β are the weights of D_i and M_i respectively.

The following is the algorithm of extracting effect statements from patent abstract.

```
Algorithm PaEffExtr:
Input: Patent abstract text, denoted as T, the set of all
ClueWords, denoted as {ClueWords}, parameters $\gamma_1$, $\gamma_2$, $\gamma_3$,
$\lambda$, $\alpha$, $\beta$ and threshold $th$
Output: The set of all annotated effect clauses, denoted
as ES
Begin
   Separate T into clauses by punctuations;
   For each clause $c_i$:
      Calculate position score $D_i$ of $c_i$ using (1);
      Segment $c_i$ into words by NLPIR, denoted as {words};
     $k_i$=0;
      For each word w in {words}:
         If w in {ClueWords}:
         $k_i$++;
      End for
      Calculate morphological score $D_i$ of $c_i$ using (2);
      $Score_i = \alpha D_i + \beta M_i$
     If $Score_i > th$:    $ES = ES \cup \{c_i\}$;
   End for
End
```

In this algorithm, firstly, the abstract text is separated into clauses with the period, comma, semicolon, colon, question mark, brackets and spaces. According to each clause's position, distribution score is calculated; secondly, segment words for each clause using NLPIR [18], count the number of clue words and calculate morphological score; thirdly, fuse distribution score and morphological scores together, and calculate the total score; finally, put all the clauses with the total score higher than the threshold into the set of effect clauses.

It is not difficult to see that the constitution of clue words set is the key point of the algorithm, since the accuracy of the set directly affects the accuracy of the algorithm. In order to ensure the integrity and usefulness of clue word set, we use the idea of iteration to collect clues words. First of all, annotate some patents manually, find out the high-frequency words, after artificial screening, keep high-quality ones as the initial set

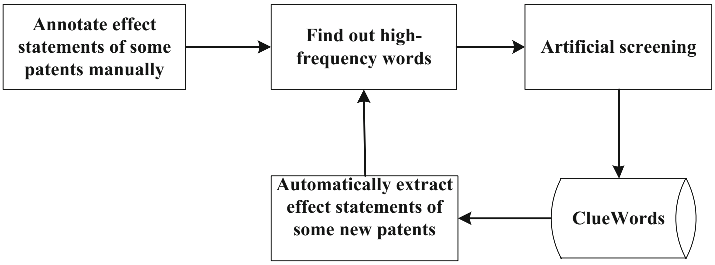
Annotate effect statements of some patents manually
Find out high-frequency words
Artificial screening
ClueWords
Automatically extract effect statements of some new patents

Fig. 1. The constitution process of clue words set

of clue words; then use the automatic extraction algorithm above to extract effect clauses of more patents, find out high-frequency words again, artificial selection, add new clue words into the initial clue words set; repeat iteratively until the clue word set arrive to a stable state. As shown in Fig. 1.

4.4 Evaluation of Algorithm

In this paper, we compare automatically annotated effect statements with those manually annotated, using two indicators: precision and recall to evaluate the effectiveness of the algorithm.

Assuming that for the *ith* patent abstract, the set of effect clauses by manual annotation is P_i, and the set of effect clauses by automatic annotation is Q_i, then, the precision and recall of our algorithm on this patent abstract are computed as the following.

$$Presicion_i = \begin{cases} \frac{|P_i \cap Q_i|}{|Q_i|} & when\ |P_i| > 0\ \ and\ \ |Q_i| > 0 \\ 1 & when\ |P_i| = 0\ \ and\ \ |Q_i| = 0 \\ 0 & when\ |P_i| > 0\ \ and\ \ |Q_i| = 0 \\ 0 & when\ |P_i| = 0\ \ and\ \ |Q_i| > 0 \end{cases} \tag{4}$$

$$Recall_i = \begin{cases} \frac{|P_i \cap Q_i|}{|P_i|} & when\ |P_i|\ > 0\ \ and\ \ |Q_i| > 0 \\ 1 & when\ |P_i| = 0 \\ 0 & when\ |P_i| > 0\ and\ \ |Q_i| = 0 \end{cases} \tag{5}$$

Generally speaking, for N patent abstracts, the precision and recall of our algorithm are computed as the following.

$$Presicion = \frac{1}{N}\sum_{i=1}^{N} Presicion_i \tag{6}$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N} Recall_i \tag{7}$$

5 Experiments

We use Java language to implement the algorithm, with 50,000 patent abstracts from Chinese universities and research institutions as the data source. In this section, the experimental results are given.

5.1 Clue Words

Table 1 exhibits part of clue words after several rounds of algorithm operation and manual screening.

Table 1. Some clue words

function	can	have	use	effect	obvious	fit for
beneficial	get	achieve	time	precision	advantage	widely
increase	low	simplify	capacity	stable	quality	side effect
outstanding	thus	shorten	range	influence	prospect	optimization
feasible	solve	treat	price	advantage	favorable	improve
important	act	solid	easy	cheap	thorough	sensibility
application	speed	high	avoid	reliability	promote	significant

5.2 Comparative Experiments

By setting different parameters and thresholds, the precision and recall of the algorithm are compared. Table 2 shows the evaluation results of 24 groups of experiments with different parameters. We can find some rules from these results.

Rule 1: We can see that when γ_1, γ_2, γ_3 values as 0.3, 0.2 and 0.5 respectively, the algorithm will get better precision and recall, since effect clauses prefer to be located in the tail the head of the abstract text.

Rule 2: The differences brought by the weights of α and β are not obvious.

Rule 3: When the threshold is improved, the precision increases, but the recall reduces.

Rule 4: When λ changes from 0.2 to 0.1, the precision increases, but the recall reduces.

Rule 5: Group 9, 23 and 24 have the best precision, and Group 15 has the best recall.

5.3 Runtime

Figure 2 shows when the parameters and threshold are set to fixed values ($\gamma_1 = 0.2$, $\gamma_2 = 0.1$, $\gamma_3 = 0.7$, $\lambda = 0.2$, $\alpha = 0.3$, $\beta = 0.7$, $th = 0.35$), the runtimes of our algorithm on different number of patent texts. It can be seen that the time complexity of our algorithm is approximately linear.

Table 2. Evaluation results of 24 groups of experiments with different parameters

Group no.	γ_1	γ_2	γ_3	λ	α	β	Threshold	Precision	Recall
1	0.333	0.333	0.333	0.2	0.5	0.5	0.3	0.612	0.659
2	0.2	0.1	0.7	0.2	0.5	0.5	0.3	0.470	0.764
3	0.3	0.2	0.5	0.2	0.5	0.5	0.3	0.565	0.765
4	0.333	0.333	0.333	0.2	0.3	0.7	0.3	0.614	0.659
5	0.2	0.1	0.7	0.2	0.3	0.7	0.3	0.565	0.765
6	0.3	0.2	0.5	0.2	0.3	0.7	0.3	0.612	0.659
7	0.333	0.333	0.333	0.2	0.5	0.5	0.4	0.678	0.494
8	0.2	0.1	0.7	0.2	0.5	0.5	0.4	0.654	0.741
9	0.3	0.2	0.5	0.2	0.5	0.5	0.4	0.736	0.675
10	0.333	0.333	0.333	0.2	0.5	0.5	0.35	0.614	0.659
11	0.2	0.1	0.7	0.2	0.5	0.5	0.35	0.488	0.764
12	0.3	0.2	0.5	0.2	0.5	0.5	0.35	0.618	0.760
13	0.333	0.333	0.333	0.1	0.5	0.5	0.2	0.437	0.773
14	0.2	0.1	0.7	0.1	0.5	0.5	0.2	0.470	0.764
15	0.3	0.2	0.5	0.1	0.5	0.5	0.2	0.401	0.778
16	0.333	0.333	0.333	0.1	0.5	0.5	0.3	0.678	0.494
17	0.2	0.1	0.7	0.1	0.5	0.5	0.3	0.495	0.731
18	0.3	0.2	0.5	0.1	0.5	0.5	0.3	0.654	0.741
19	0.333	0.333	0.333	0.2	0.4	0.6	0.3	0.612	0.659
20	0.2	0.1	0.7	0.2	0.4	0.6	0.3	0.618	0.760
21	0.3	0.2	0.5	0.2	0.4	0.6	0.3	0.565	0.765
22	0.333	0.333	0.333	0.2	0.4	0.6	0.4	0.678	0.494
23	0.2	0.1	0.7	0.2	0.4	0.6	0.4	0.736	0.675
24	0.3	0.2	0.5	0.2	0.4	0.6	0.4	0.736	0.675

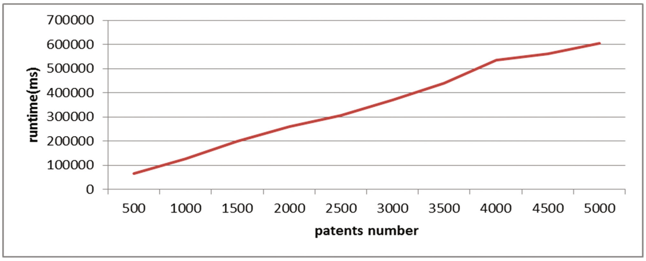

Fig. 2. The runtime of the algorithm

6 Conclusion and Future Work

In order to reduce the burden of patent annotators, this paper presents an automatic extraction algorithm of effect statements in Chinese patent abstracts. This algorithm uses distribution and morphological characteristics of effect statements, construct a clue words thesaurus, and use scoring method to extract effect statements automatically. The algorithm is simple and direct, and has satisfying experimental results. It can also be extended to the automatic annotation of other patents information, such as technical words, coordinative phrases, and so on.

Acknowledgments. This paper was supported by Research Foundation for Advanced Talents of Hubei University of Technology (No. BSQD12131), the Fundamental Research Funds for the Young Teachers' Innovation project of Zhongnan University of Economics and Law (No. 2014147), the Educational Commission of Hubei Province of China (No. D20151401) and the Green Industry Technology Leading Project of Hubei University of Technology (No. ZZTS2017006).

References

1. Zhang, L., Li, L., Li, T.: Patent mining. ACM SIGKDD Explor. Newsletter **16**(2), 1–19 (2015)
2. Fan, Y., Hongguang, F.U., Wen, Y.: Patent information clustering technique based on latent Dirichlet allocation model. J. Comput. Appl. (2013)
3. Jun, S., Sang, S.P., Dong, S.J.: Technology forecasting using matrix map and patent clustering. Ind. Manag. Data Syst. **112**(5), 786–807 (2012)
4. Choi, S., Jun, S.: Vacant technology forecasting using new Bayesian patent clustering. Technol. Anal. Strateg. Manag. **26**(3), 241–251 (2014)
5. Sharma, A.: A Survey On Different Text Clustering Techniques For Patent Analysis. Esrsa Publications (2012)
6. Wu, J.L., Chang, P.C., Tsao, C.C., et al.: A patent quality analysis and classification system using self-organizing maps with support vector machine. Appl. Soft Comput. **41**, 305–316 (2016)
7. Xia, B., Baoan, L.I., Lv, X.: Research on patent document classification based on deep learning. In: International Conference on Artificial Intelligence and Industrial Engineering (2016)
8. Noh, H., Jo, Y., Lee, S.: Keyword selection and processing strategy for applying text mining to patent analysis. Expert Syst. Appl. **42**(9), 4348–4360 (2015)
9. Nonaka, H., Kobayahi, A., Sakaji, H., et al.: Extraction of the effect and the technology terms from a patent document. In: International Conference on Computers and Industrial Engineering, pp. 1–6. IEEE (2010)
10. Nonaka, H., Kobayashi, A., Sakaji, H., et al.: Extraction of effect and technology terms from a patent document (theory and methodology). J. Jpn. Ind. Manag. Assoc. **63**, 105–111 (2012)
11. He, Y., Li, Y., Meng, L.: A new method of creating patent technology-effect matrix based on semantic role labeling. In: International Conference on Identification, Information, and Knowledge in the Internet of Things, pp. 58–61. IEEE (2015)
12. Chen, Y.: Research of patent technology-effect matrix construction based on feature degree and lexical model. New Technology of Library & Information Service (2012)
13. Hou, T., Lv, X.Q., Xu, L.P.: Chinese patent efficacy phrase recognition. Appl. Mech. Mater. **743**, 510–514 (2015)

14. Chen, X., Deng, N.: A semi-supervised machine learning method for Chinese patent effect annotation. In: International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp. 243–250. IEEE Computer Society (2015)
15. Chen, X., Peng, Z., Zeng, C.: A co-training based method for Chinese patent semantic annotation. In: ACM International Conference on Information and Knowledge Management, pp. 2379–2382. ACM (2012)
16. Deng, N., Chen, X.: Automatically generation and evaluation of stop words list for Chinese patents. Telkomnika **13**(4), 1414 (2015)
17. Deng, N., Chen, X., Li, D.: Intelligent recommendation of Chinese traditional medicine patents supporting new medicine's R&D. J. Comput. Theor. Nanosci. **13**, 5907–5913 (2016)
18. http://ictclas.nlpir.org/newsDetail?DocId=387