Chapter 4
Query Intent Understanding

Zhicheng Dou and Jiafeng Guo

Abstract Search engines aim at helping users find relevant results from the Web. Understanding the underlying intent of queries issued to search engines is a critical step toward this goal. Till now, it is still a challenge to have a scientific definition of query intent. Existing approaches attempting to understand query intents can be classified into two categories: (1) query intent classification: mapping queries into categories and (2) query intent mining: finding subtopics covered by the queries. For the first group of work, the mapping between queries and categories can be conducted in various ways, including classifying based on navigational, informational, or transactional intent, based on geographic locality, temporal intent, topical categories, or available vertical services. For query intent mining, the output can be a list of explicit subqueries, or some implicit representation of subintent, such as a list of document clusters, a list of entities, etc. In this chapter, we will introduce these query intent prediction approaches in detail.

4.1 Introduction to Query Intent Understanding

Search engines aim at helping users find relevant results from the Web. In most existing Web search engines, users' information needs are represented by simple keyword queries. Studies have shown that the vast majority of queries issued to search engines are short, usually comprised of two to three keywords [19, 28, 45, 52, 53]. How to precisely understand the complex search intent implicitly represented by such short queries is a critical and challenging problem and has received much attention in both IR academic and industry communities.

Z. Dou (✉)
Renmin University of China, Beijing, China
e-mail: dou@ruc.edu.cn

J. Guo
Chinese Academy of Sciences, Beijing, China
e-mail: guojiafeng@ict.ac.cn

© Springer Nature Switzerland AG 2020

Y. Chang, H. Deng (eds.), *Query Understanding for Search Engines*,
The Information Retrieval Series 46, https://doi.org/10.1007/978-3-030-58334-7_4

Query intent itself is an ambiguous word, and it is still a challenge to have a scientific definition of query intent. Intent itself means the perceived need for information that leads to a search, but how to describe or classify the need is still in an exploratory stage. Till now, different kinds of query intent understanding tasks have been explored toward discovering the implicit factors related to real user information needs. These tasks include but are not limited to identifying the type of search goals and demanded resources required by a user, identifying the topical categories a query belongs to, selecting vertical services a query might be relevant to, and mining subintents for an ambiguous or broad query. Basically, query intent understanding is mainly for the purpose of recovering the hidden aspects that belong to the original user information need but is lost within the short and simple keyword queries issued to search engines.

Existing approaches attempting to understand query intents can be roughly grouped into two categories as follows:

Intent classification: This is basically a task that maps queries into categories. The mapping between queries and categories can be conducted in various ways, such as classifying based on user goals like navigational, informational, or transactional intent, classifying based on topical categories, classifying based on vertical services, classifying based on geographic locality, or classifying based on temporal intent.

Intent mining: The task is mainly for broad or ambiguous queries. It aims to find subtopics covered by a query. The output can be a list of explicit subqueries, or some implicit representation of subintent such as a list of document clusters, a list of entities, etc.

In this chapter, we will introduce existing query intent understanding approaches in detail.

4.2 Intent Classification Based on User Goals

A major difference between Web search and classic IR (information retrieval) lies in that users' search need/goal is no longer restricted to acquiring certain information—they might search to locate a particular site or to access some Web services. Therefore, the first type of query intent understanding tasks we discuss is identifying the underlying goal of a user when submitting one particular query. More specifically, it aims to classifying user goals into navigational, informational, transactional, etc. For instance, when a user issues the query "amazon", he or she could be trying to reach the specific website http://www.amazon.com; while a user submitting "Olympic history" is most likely to be interested in finding information on that topic but not concerned about the particular website. The query "adobe photoshop download" might indicate that the user is finding a Web page where he or she can find a link to download the desired software. In this case, the query is more likely to be an transactional query, other than informational or navigational.

4.2.1 Taxonomies of User Goals

Basically, user goals can be classified based on the type of demanded resources users are seeking for by issuing a query. Several taxonomies of user goals have been proposed since Broder [10] introduced this concept. In the first part of this subsection, we will briefly introduce these taxonomies.

4.2.1.1 Broder's Intent Taxonomy

The first and most popular taxonomy of query intent (here intent means user goal) on the Web was proposed by Broder [10]. According to Broder, there are three classes of queries: informational, navigational, and transactional, which are introduced in detail as follows.

Navigational Navigational intent means that a user's immediate intent is to reach a particular website for browsing. The website could be a website the user has visited it in the past. The user uses a navigational query to reach this website because it is more convenient for his or her to input a short navigational query other than typing the URL. A user may also issue a navigational query to find a website he or she never visited in the past, but she assumes that there should be such a website. Example navigational queries are

- `Renmin University of China`. The target website of the user who submits this query is likely to be http://www.ruc.edu.cn, the homepage of Renmin University of China.
- `jd.com`. Users may want to use this URL-like query to directly reach the website http://www.jd.com.
- `apple store`. Most users might use this query to find http://store.apple.com.

As shown by the previous examples, the most typical navigational queries are those homepage-finding queries. A navigational query has usually one "perfect" result, which is exactly the website the user is looking for. But in some rare cases, a navigational query could be ambiguous, and different users might use the same query to find their particular websites. For example, a user might use "aa" to reach https://www.aa.com, whereas another might use the same query to navigate to http://www.aa.org.

Informational For informational queries, the user wants to obtain some information assumed to be available on the Web. The information could be present on one or multiple Web pages. Broder emphasized that the information could be found on these Web pages in a *static* form, which means that "no further interaction is predicted, except reading" [10]. Example informational queries include

- `how to cook beef`. Users are finding more ways to cook beef.
- `Beijing tourist attractions`. Users use this query to find a list of tourist attractions in Beijing and detailed introduction to them.
- `deep learning`. Users might use this query to learn information about deep learning, such as the definition, architectures, algorithms, or applications.

Transactional The goal of a transactional query is to find a Web page where he or she can then perform some interactive tasks such as downloading a software, listening to music, or playing a game online. Example transactional queries are

- `7zip download`. The goal is to find a link for downloading the file compression software 7zip.
- `currency converter`. Users use this query to find a currency converter and then calculate live currency and foreign exchange rates with this currency converter.

Broder studied the statistics of these types of queries by doing a survey of 3,190 valid AltaVista users. The survey results indicated that about 24.5% of queries are navigational queries. He also found that it is not easy use a single question to distinguish between transactional and informational queries by the survey. Alternatively, by asking users whether they are shopping or want to download a file, he estimated that at least 22% of queries are transactional queries. Broder further manually assessed 400 queries from the AltaVista log, and found about 20% are navigational, 48% are informational, and 30% are transactional queries, leaving 2% of queries undetermined in their intents.

4.2.1.2 Rose and Levinson's Taxonomy

Rose and Levinson [47] further improved Broder's intent classification and proposed a hierarchy of query goals with three top-level categories. They developed a framework for manual classification of search goals and introduced subcategories for some classes. Specifically, Rose and Levinson divided informational intent into five sub classes as follows:

- **Directed**: directly answering open or closed questions,
- **Undirected**: undirected requests to simply learn more about a topic,
- **Advice**: requests for advice,
- **Location**: the desire to locate something in the real world,
- **List**: simply getting a list of suggestions for further research.

At the same time, they replaced the transactional intent with the "resource" intent, which represents the goal of obtaining something other than information from the Web. The resource intent is comprised of four specific interactive tasks including "download," "entertainment," "interact," and "obtain."

Table 4.1 Intent taxonomy proposed by Rose and Levinson [47]

Search goal	Minor classes	Percentage	Broder's
Navigational	/	13–16%	Navigational
Informational	Directed, undirected, advice, locate, list	61–63%	Informational
Resource	Download, entertainment, interact, obtain	21–27%	Transactional

Rose and Levinson [47] studied the distribution of different types of queries by manually classifying queries from AltaVista query logs. They found that about 61% to 63% of queries are informational queries, and 13% to 16% are navigational. More details are shown in Table 4.1.

4.2.1.3 Taxonomy Proposed by Baeza-Yates et al.

Different from the above two taxonomies that classify queries into navigational, informational, and transactional (or resource), Baeza-Yates et al. [4] established a slightly different classification system of user goals. They classify queries into **Informational**, **Not informational**, and **Ambiguous**. Based on their definition, the informational intent is similar to the informational intent defined by Broder [10] and Rose and Levinson [47]. Differently, they merged navigational queries and transactional queries into a single category: "Not informational" queries, because both types of queries are issued to find other resources other than information on the Web. Baeza-Yates et al. further introduced the third category: ambiguous queries. An ambiguous query means that its user goal cannot be easily inferred based on the query string without additional resources. More information about query ambiguity will be introduced in Sect. 4.4.

Baeza-Yates et al. [4] studied the distribution of queries based on a log sample containing about 6,000 queries from the Chilean Web search engine TodoCL.[1] They manually classified these queries and found that 61% of queries are informational queries, 22% are not informational queries, and about 17% are ambiguous.

4.2.1.4 Taxonomy Proposed by Jansen et al.

Jansen, Booth, and Spike [30] presented a three-level hierarchical taxonomy based on existing taxonomies, with the top most level being informational, navigational, and transactional. They also provided a comprehensive reviews and evaluation of the different query intent taxonomies proposed in the literature by aligning prior work to their categorizations. Their studies showed that about 81% of queries are informational, 10% are navigational, and about 9% are transactional queries, based

[1] TOdoCL, http://www.todocl.com.

Table 4.2 Distribution of query intents in existing studies

Intent type	Broder	Rose and Levinson	Baeza-Yates et al.	Jansen et al.
Navigational	20%	13%–16%	/	10%
Informational	40%	61%–63%	61%	81%
Transactional	30%	/	/	9%
Resource	/	21%–27%	/	/
Not informational	/	/	22%	/
Ambiguous	/	/	17%	/

on automatic and manual analysis over the Dogpile[2] search engine transaction log. Note that the proportion of informational queries is much higher than those reported in previous works. They believed that the variation in the reported percentage may be related to the small-size samples used in prior studies and the power log distribution of Web queries. Readers who are interested in this taxonomy can read [30] for more details.

4.2.1.5 Summarization

We summarize the major intent types defined in existing studies, together with the distributions of queries belonging to these intents according to the original studies. The statistics is shown in Table 4.2. The table indicates that although a large percentage of queries issued to search engines are for information seeking (informational queries), there are still many queries that are issued for other intents, such as seeking a particular website or performing an interactive task.

All these studies have provided deeper understanding on users' search goals with more specific and detailed definitions on intent taxonomy. However, from a review of the existing literature, Broder's taxonomy is the most widely adopted one in automatic query intent classification work probably due to its simplicity and essence. Besides, it is worth to note that not the full taxonomy of Broder has been utilized in all the intent classification works. There are studies trying to identify navigational and informational queries [32, 34], or differentiating transactional or navigational queries from the rest. Different features have been designed according to the specific classification tasks as we will show in Sect. 4.2.3.

4.2.2 Methods Used for Predicting User Goals

Although various kinds of taxonomies are proposed to classify different underlying goals of the user when submitting one particular query, a common premise is that

[2] http://www.dogpile.com/.

when users use search engines to seek information, their goals are diverse. With the classification of different intentions driving user queries, search engines can utilize different ranking mechanisms to support different types of queries and to improve user experience. For example, for software downloading queries, search engines can provide a direct download link in the search result page.

Early work on query intent classification performed manual classification to establish the intent taxonomy [10, 47] and verified the feasibility of automatic intent classification [34]. Labeling tools with carefully designed questionnaire were utilized to facilitate the manual classification process. Later, automatically identifying such intents became the mainstream in this research community, starting from heuristically constructed classifiers. In this section, we will briefly review these approaches. As we just mentioned, although different taxonomies have been proposed as we introduced in the previous section, Broder's taxonomy is most received by IR community. Furthermore, Broder's study has shown that transactional queries are usually hard to be identified from navigational queries and informational queries. Hence, most effort on automatically identifying user goals focused on simply dividing queries into navigational and informational.

User goals can be automatically identified by either unsupervised methods (rule-based methods) or supervised learning-based methods. For unsupervised methods, one or multiple rules are manually created for identifying query types. For example, Kang et al. [32] utilized a linear function to generate a score based on four measures to decide the query intent. Lee et al. [34] adopted a similar linear combination approach and used the threshold derived from the goal-prediction graph to classify query intents. Brenes et al. [8, 9] ranked queries based on three types of features to detect navigational queries. Jansen et al. [29, 30] implemented an automatic classifier based on handcrafted rules by identifying the linguistic characteristics of queries with respect to different intents (these features will be introduced in Sect. 4.2.3.1). All of these methods relied on "*ad hoc*" thresholds and parameters.

To avoid such heuristics, some researchers turned to supervised learning-based methods, and different models have been used in existing approaches. Among these models, linear regression, SVM, and decision tree are widely used. Linear regression and decision tree can generate interpretable models and illustrate the usefulness of each feature studied, while SVM is shown to be useful for processing high-dimensional vectors, especially those text-based features. For example, Kang and Kim [32] and Lee et al. [34] used the linear regression model to classify queries. Nettleton et al. [42] employed Kohonen self-organized maps (SOM) and C4.5 decision trees to classify user sessions into informational, navigational, and transactional. Liu et al. [37] also used C4.5 decision tree model for query intent classification. Baeza-Yates et al. [4] and Lu et al. [40] employed SVM for intent classification.

To better model users' search sessions, Hu et al. [25] proposed to use skip-chain Conditional Random Field (CRF) to predict commercial query intent. The skip-chain CRF can model the correlation between nonconsecutive similar queries in users' search sessions via skip edges to improve the prediction accuracy. Similarly, Deufemia et al. [18] employed both CRF and Latent Dynamic Conditional Random

Field (LDCRF) to model sequential information between queries within a user session and showed that CRF can achieve better performance than SVM on informational query identification. Multitask learning has also been used in query intent classification. In [7], Bian et al. proposed to learn both ranking functions and query intent classifier simultaneously. A logistic model is utilized to predict the probability of query intents. The ranking function jointly learned with query categorization was demonstrated to be more effective than that learned with predefined query categorization.

Furthermore, Lu et al. [40] compared several machine learning methods, including naive Bayes model, maximum entropy model, SVM, and stochastic gradient boosting tree (SGBT), for navigational query identification. They found that SGBT coupled with linear SVM feature selection is most effective. Zamora et al. [64] studied decision trees, SVM, and ensemble methods for query intent classification with respect to the taxonomy of Broder. They found the use of ensembles allows to reach significant performance improvements.

Beside these classification models, Baeza-Yates et al. [4] employed Probabilistic Latent Semantic Analysis (PLSA), an unsupervised method to cluster queries into informational, not informational, and ambiguous categories. They also applied the supervised learning method SVM and found that the combination of supervised and unsupervised learning is a good alternative to find user's goals, rather than the sole use of each method.

4.2.3 Features

As discussed, user goals can be identified by either unsupervised methods (rule-based methods) or supervised learning-based methods. Both types of methods rely on one or multiple well-designed features, which reflect characteristics of different types of queries. There are a large number of features proposed by existing works. These features, can be extracted from query string itself, document corpus, query logs, anchor texts, or summaries of top search results. Some features were proposed according to specific classification tasks, such as for classifying intent into navigational/navigational/transactional, into navigational/non-navigational, or into informational/non-informational. We think that most features can be assumed to be independent of the taxonomy used, although they are originally proposed for a specific classification task. Hence here we mainly categorize the features into three groups according to the data resources and the types of the features:

- **Features extracted from query strings**: linguistic features defined based on the surface strings of the query;
- **Features extracted from the corpus**: features defined on the corpus to be retrieved or the top retrieved documents, typically using document content, anchor texts, or URL information.

- **Features extracted from query log**: features defined on the user interaction logs recorded by search engines/toolbars, typically using information such as click-through, sessions, and eye/mouse movement.

In the remaining part of this section, we will briefly introduce some commonly used features within each category. At the end of the section, we will briefly summarize where the features are used and what classification task they are used for.

4.2.3.1 Features Extracted from Query Strings

The simplest features used for identifying query intent are linguistic characteristics of query terms or query strings, for example, whether the query string contains specific characters, URLs, or entity names. Jansen et al. [29, 30] tried to classify query intent into informational, navigational, and transactional based on characteristics of queries and query terms. They used some simple features extracted from query strings, such as query length (they assumed that a navigational query has less than three terms). They identified key characteristics of different categories of queries based on an analysis of queries from three different Web search engines. For example, navigational queries are queries containing company/business/organization/people **names**, or queries **containing domain suffixes**. Transactional queries are identified by checking whether queries contain **specific terms** (for example, "lyrics," "download," "images," "audio," "buy" for transactional intent, "ways to," "how to," "list" for informational). A simple rule-based classifier was implemented to identify query categories based on the above characteristics. They then used this classifier to categorize a million real queries and found that more than 80% of Web queries are informational, with about 10% each being navigational and transactional.

Kang and Kim [32] also used linguistic features. They assumed that navigational queries are usually proper names, whereas some informational queries may include a **verb**. They simply classify the queries that have a verb (except the "be" verb) into informational queries.

4.2.3.2 Features Extracted from the Corpus

Kang and Kim [32] employed the WT10g[3] dataset to build two document subsets, namely DBHOME and DBTOPIC, to identify intent types. DBHOME is comprised of those documents acting as entry points for a particular website within WT10g, while DBTOPIC includes the remaining Web pages in WT10g. Kang and Kim pro-

[3] http://ir.dcs.gla.ac.uk/test_collections/wt10g.html.

posed several search corpus-based features that consider the following information contained in both sets:

- the distribution of query terms in both subsets,
- the mutual information of query term pairs in both subsets.

They further assumed that terms of navigational queries appear in titles and anchor texts more frequently than informational queries. They utilized the probability that a query appear in anchor text and page titles as a feature for predicting user goals. They combined the above three types of features and the query string-based feature we just introduced (i.e., containing verb) to classify query intent into informational and.

Kang [31] then proposed to explore hyperlink information for transactional intent detection. Specifically, he clustered hyperlinks according to the extension of a linked object (e.g., site, music, or file) with the assumption that some types of hyperlinks are more likely to be linked to transactional activities (for example, if the linked object is a binary file, its possible activity is downloading). He then extracted cue expressions (i.e., short definition or explanation) for each hyperlink type based on titles and anchor texts. Based on this information, Kang proposed a new set of features called *link scores* for each query. The basic idea was to calculate the proportion of candidate expressions (i.e., the whole expression, the first and last term, and the first and last biterm of the query) in the collection of cue expressions that represent each hyperlink type. The experimental dataset consisted of 495 navigational and informational queries from TREC and 100 transactional queries manually extracted from a Lycos[4] log file. Using the proposed features as well as those in [32], he achieved the overall performance of 78% in both precision and recall for the identification of transactional queries.

Lee et al. [34] defined anchor-link distribution in the search corpus as a feature for intent classification. They checked the destinations of the links with the same anchor text as the query. For a navigational query, a single authoritative website exists (i.e., a dominating portion of links with the query as the anchor text point to this website). On the contrary, for an informational query, because of lack of a single authoritative site, the links with the query as anchor text may point to a number of different destinations. Lee et al. located all the anchor links that have the same text as the query, extracted their destination URLs, counted the number of links for each distinct URL, sorted the URLs in the descending order of link numbers, and finally calculated the distribution of links over these distinct URLs. The anchor-link distribution of a navigational query is expected to be highly skewed toward the most frequent URL, whereas the anchor-link distribution for an informational query should be more flat. They used mean, median, skewness, and kurtosis to measure the skewness of anchor-link distribution and used them as features for query intent classification. Anchor-link distribution can be considered as an alternative of query-click distribution (which will be introduced later) when

[4]http://lycos.com.

click-through data is unavailable or sparse. Liu et al. [37] and Lu et al. [40] also used the anchor-link distribution for identifying navigational queries.

Herrera et al. [24] studied search corpus features (including anchor text-based features and page content-based features). Beside those previously proposed features, they included the use of some new features. One of the new features is based on the idea that statistics about the occurrence of the query terms across different domains are useful for determining the user goal. They used this assumption to include two new features, namely density of domains in the top similar anchor texts and density of domains in the top similar texts, which compute the ratio of distinct domains in top K answers in top similar anchor texts and top retrieved pages, respectively. Another feature is the popularity of the query. They utilized the WT10g query set the same as [32] and additional 600 queries from the WBR03 collection, 200 queries for each intent category. By using all the features, they achieved an accuracy of 82.5% on WBR03 queries and 77.67% on WT10g queries. They showed that the query popularity feature is effective when combined with other features, increasing their discriminative nature.

4.2.3.3 Features Based on Query Log

Query log is one of the effective data sources for search ranking and intent understanding. It has been well utilized in existing works on query goal identification. Lee et al. [34] and Liu et al. [37] investigated the problem of separating navigational queries from informational based on click-through data. Both approaches computed the click distribution from click-through data for each query. Given a query, its click distribution is constructed as follows:

1. count the times each document is clicked by all users under the query;
2. sort all clicked documents in the descending order of the total number of clicks made on the documents by all users;
3. normalize click frequencies so that all values add up to 1 and get the distribution.

Basically, similar to the anchor-link distribution we just introduced, if the click distribution of a query is highly skewed toward one or just a few domains or Web pages, the query is more likely to be a navigational query. In contrast, when the click distribution is relatively flat, the query tends to be informational. To summarize click distribution into a single numeric feature that captures how skewed the distribution is, different statistical measurements, such as mean, median, skewness, and kurtosis, can be used. Click entropy, which was proposed by Dou et al. [19, 20], can also be used to quantifying a click distribution. Wang and Agichtein [60] revisited the classification problem with respect to clear (navigational)/informational/ambiguous proposed by Baeza-Yates et al. [4]. They proposed entropy-based metrics of the click distributions of individual searchers, which is better than entropy of all result clicks of a query in distinguishing informational and ambiguous queries. They also involved domain entropy as a backoff to the URL entropy to deal with the sparsity problem. Using the 150 manually labeled queries from MSN search query log, they showed user-based click entropy features can improve the classification performance as compared with overall entropy features.

In addition, Lee et al. assumed that navigational queries are usually associated with fewer clicks than informational ones; hence, they used the average number of clicks of a query as another feature to identify navigational queries. Liu et al. [37] also observed that navigational queries usually have fewer clicks than informational or transactional queries. Differently, they use "n Clicks Satisfied (nCS)" to quantify this. nCS is the proportion of sessions containing a given query in which the user clicked at most n results. They further assumed that users tend to click on the top results of navigational queries. Based on this, they proposed to use "top n Results Satisfied (nRS)," the proportion of sessions containing a given query in which the user clicked at most top n results. Given a small n value (e.g., two), navigational queries tend to have higher nCS and nRS values than informational or transactional queries.

Brenes and Gayo-Avello [8] proposed three user log features, each associated with a Navigational Coefficient (NC), to identify navigational queries. The first NC is the rate of visits to the most visited result in the query. It is equal to the click probability of the rank no. 1 result (i.e., the maximum click probability) in the click distribution we have introduced. The second NC is defined as $1 - \frac{\text{number of distinct results}}{\text{number of visits to all results}}$. The third and last value, *percentage of navigational sessions*, computes the ratio of one-query one-click sessions to all the sessions containing that query. Each NC was then used to rank the queries from AOL search logs, and only case studies were conducted for evaluation.

Nettleton et al. [42] used number of clicks, click position, and used browsing time on clicked documents as features for predicting user goals. Deufemia et al. [18] introduced several new interaction features based on user behaviors during the exploration of Web pages associated to the links of the SERP. They not only considered the absolute and effective dwell time on a Web page but also measured the amount of reading of a Web page and the number of words during the browsing. There were also some interaction features designed for transactional queries, such as *AjaxRequestsCount* that represents the number of AJAX requests originated during browsing. The basic assumption is that capturing interaction features on specific portions of Web pages conveys a better accuracy in the evaluation of user actions. They collected 129 labeled search sessions from 13 subjects for evaluation. Using the proposed interaction features together with traditional query, search, and context features, they achieved 0.84, 0.88, and 0.86 for transactional, informational, and navigational query identification, respectively. They also demonstrated the effectiveness of the transactional interaction features for transactional queries.

Guo and Agichtein [22] explored mouse movements for inferring informational and navigational intents. The features included average trajectory length, average vertical range, and average horizontal range. Based on 300 labeled queries from the MSN search engine, they showed that using these simple features can achieve an accuracy of 70.28% for intent classification.

4.2.3.4 Features Leveraging Multiple Sources

Baeza-Yates et al. [4] proposed to use terms from the documents clicked by the query to construct the feature vector and group the queries into clusters. Using a dataset of 6042 manually labeled queries according to informational, non-informational, or ambiguous intentions, they constructed feature vectors from a query log from the Chilean Web search engine TodoCL.[5] Evaluation results demonstrated that such term-based features are good at detecting informational queries (approximately 80% precision with recall above 80%) but less effective on non-informational (close to 60% precision with 40% recall) and ambiguous queries (less than 40% precision with recall lower than 20%). In [41], Mendoza and Zamora further extended this vector representation by considering the time users take to review the documents they select, leading to *tf-idf-time* and *tf-idf-pop-time* weighting schemes. The basic idea is that the time spent in each query differs by query intent (for example, an informational query may take more time for the user to review the result pages). Based on 2000 labeled queries, they showed that vector representation based on *tf-idf-time* weighting scheme is the most effective (above 90% in F-measure) in identifying informational/navigational/transaction intents as compared with that based on *tf-pop* and *tf-idf-pop-time* schemes.

Liu et al. [39] proposed to leverage Web page forms to generate useful query patterns for transactional query identification. Specifically, they first analyzed the distribution of form clicks and obtained a group of high-quality transactional queries by mining toolbar log. With these transactional queries as training data, they matched them with the information contained in forms to help generalize these queries into patterns. These transactional query patterns along with a confidence score were used as basic features to classify new queries. Note that in this work, the authors used both corpus-based features (Web page forms) and query log-based features (toolbar log).

4.2.3.5 Summary of Features Used

Table 4.3 summarizes some main features used in existing approaches. Brenes et al. [9] did a survey and evaluation of query intent detection methods. They found that the combination of features extracted from query terms, anchor text, and query log performed the best. Beside these approaches, there also exists some effort on feature engineering over a large number of features for query intent identification. For example, Lu et al. [40] studied both search corpus and user log features for navigational query detection. For each query, the top 100 URLs were recorded and 100 query–URLs were generated for features construction. For each query–URL pair, they extracted a total of 197 features, among which 29 features are query log features using click information, and the rest are search corpus features based on

[5] http://www.todocl.com.

Table 4.3 Features used for query intent classification

Source	Feature	Work
Query string	Containing entities (company, business, organization, people names)	Jansen et al. [30]
	Containing domain suffixes	
	Containing clue words (lyrics, download, image, etc.)	
	POS, containing verb	Kang and Kim [32]
Corpus	Anchor-link distribution (mean, median, skewness, kurtosis, etc.)	Lee et al. [34], Liu et al. [37], Lu et al.[40]
	Query term distribution of subdocument sets (HOME and TOPIC), etc.	Kang and Kim [32],
	The usage rate of query term as anchor texts and page titles	Kang [31]
	Link scores	Kang [31]
Query log	Average number of clicks	Lee et al. [34], Liu et al. [37], Nettleton et al. [42]
	Click distribution (mean, median, skewness, kurtosis, etc.)	Lee et al. [34], Liu et al. [37], Lu et al. [40]
	Click probability of the most clicked result, i.e., click distribution (max)	Brenes and Gayo-Avello [8], Lu et al. [40]
	n Clicks Satisfied (nCS)	Liu et al. [37]
	top n Results Satisfied (nRS)	
	Click entropy	Dou et al. [19], Lu et al. [40]
	Click position	Nettleton et al. [42]
	Browsing time	
	Mouse movements	Guo et al. [22]

URL itself and anchor texts pointed to the URL. Feature integration operators such as normalized ratio, mean, and entropy were then utilized to calculate statistics of the raw features. In this way, the combination of selected features yield the best classification result.

4.2.4 Summary

Query intent classification based on user goals attempts to categorize the underlying goal of users' search. Broder's taxonomy and its simplified variants have been widely adopted as the major intent taxonomies. Researchers have developed different types of features in order to enrich the query representation for the classification tasks, from simple query string features using surface term characteristics, to corpus-based features leveraging Web content information, to query log features capturing user interactive behaviors. This line of research started in early 2000 and reached its peak in around 2008–2009, with diverse models and features emerging in the research community. However, the lack of a benchmark dataset devoted to the task makes it difficult to fairly compare existing work. One may refer to the work from Brenes et al. [9], which partially addressed this problem by comparing several previous methods based on a large query set (6624 queries) from MSN Query Log.

4.3 Vertical Intent Classification

With the emergence of numerous vertical search services (e.g., job search, product search, image search, map search, news search, weather search, or academic search), it is becoming popular in search engines to present aggregated results from multiple verticals through the standard general Web search interface. This is so-called aggregated search or federated search. An example aggregated search result page from Bing search engine (http://www.bing.com) is shown in Fig. 4.1. A customized region containing latest weather forecast information of Beijing city is directly shown in the search result of query "Beijing weather." Directly showing this more specialized answer region in SERP will benefit most users, hence they do not need to spend extra effort on opening normal Web search results to browse the detailed information again. Furthermore, with this kind of aggregated search, users do not have to identify his or her intent in advance and decide which vertical service to choose to satisfy his or her intention. This usually reduces user efforts and hence can greatly improve user satisfaction.

At the same time, irrelevant vertical results within the search engine result page (SERP) may disturb users. For example, providing image search results in SERP for query "Beijing weather," or displaying weather vertical results for query "weather forecasting method" is useless or even detrimental to user experience. Therefore, it is critical to have query vertical intent classifiers in a general or aggregated search engine that can predict whether a query should trigger respective vertical search services. This is also called vertical selection problem [3, 25]. Note that a query may implicitly cover more than one intent or vertical.

4.3.1 Topical Intent Classification

Some verticals are genre specific [2]. Therefore, some prior work in topical intent classification is relevant to vertical selection. The main target of topical intent classification is to classify a query into a ranked list of n categories (e.g., assigning the query "Transformers" to the category "Entertainment/Movies" and "Entertainment/Games").

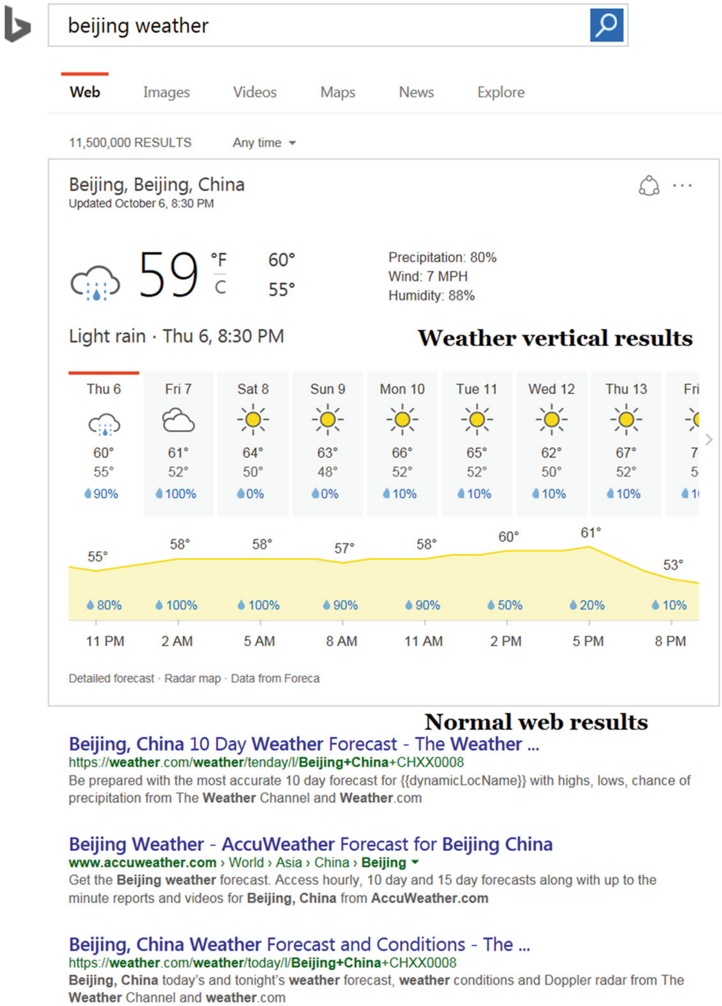

Fig. 4.1 An example aggregated search result page for the query "Beijing weather" from Bing (http://www.bing.com). A region containing latest weather forecast information of Beijing city is shown in the search result. Users can directly get this information without extra effort for viewing normal search results or opening corresponding vertical search engines

The main challenge of classifying Web queries is the sparseness of query features due to the limitation of information provided by short Web queries. To solve this problem, most topical query classification approaches leverage external data sources, in addition to the original query strings, to enrich features. One typical way is to extract features from search engine results, including the document content, titles, URLs, and snippets. For example, Shen et al. [50] used the titles, the snippets, and the full plain text of the documents returned by search engines and ODP taxonomies[6] to generate textual features for classifying queries into 67 target topical categories, based on support vector machine (SVM) classifiers. Broder et al. [11] used retrieved search results to classify queries into a commercial taxonomy comprised of approximately 6000 nodes within the sponsored search environment. Given a query, they issued the query to a general Web search engine, classified the returned Web pages, and then used the page classification results to classify the original query. Beitzel et al. [6] found that a classifier trained using snippets from the retrieved documents performs merely 11% better than using only query lexical features (mainly query terms).

In addition to the work primarily focusing on enriching feature representation, some other approaches aim at obtaining more training data from query logs by semi-supervised learning. For example, Beitzel et al. [5] leveraged unlabeled data to improve supervised learning. They developed a rule-based automatic classifier produced using selectional preferences mined from the linguistic analysis of a large-scale query log. They used this unsupervised classifier to mine a large number of unlabeled queries from query logs as training data, together with some manually classified queries, to improve the supervised query classification models.

As the Query Topic Classification task has been discussed in Sect. 4.3 of the Query Classification chapter, we will not cover those again in this chapter.

4.3.2 Vertical Intent Classification

In addition to detecting the topical categories, some other vertical intent classification methods have been proposed by utilizing more resources, which are summarized as follows.

(1) Content of vertical corpus. Vertical intent can be classified by evaluating whether the query is relevant to the content of each vertical or whether the vertical can return sufficient amount of information.
(2) Query strings. Vertical services specialize on identifiable domains and types of media. This enables users to possibly express interest in vertical content explicitly [2], using keywords such as "news" for the news vertical or "weather"

[6] http://www.dmoz.org.

for the weather vertical. Therefore, another potentially useful source of evidence for vertical intent classification is the query string itself.

(3) Characteristic of normal Web search results. Characteristic of search results returned from the normal search service (i.e., the Web vertical) is also helpful for detecting vertical type of the query. For example, if many shopping websites are returned for a query, it is likely that the query has a commercial intent.

(4) User behaviors on verticals and the aggregated search page. Some verticals have a search interface through which users directly search for vertical content. For example, Bing search engine (http://www.bing.com) has a separated search service (http://www.bing.com/images) for image vertical and http://www.bing.com/news for news vertical. Vertical intent of a query can be estimated by evaluating whether users actively use this query in the vertical, or other user behaviors gathered from these search services. At the same time, some users prefer the default Web search interface, other than separated vertical services. Furthermore, some verticals do not have a separated interface. The rich user behaviors made on the default search page can be utilized for vertical intent classification. For example, whether users click image answers more frequently than normal Web pages for query "tom cruise" is an important implicit feedback for judging the image vertical intent of "tom cruise."

Details of the features will be introduced in the remaining part of this section.

4.3.2.1 Corpus-Based Features

As Arguello et al. introduced, we may view vertical intent classification (vertical selection) analogous to resource selection in federated search [2, 3], if we consider verticals as external collections. Corpus-based features are derived from document rankings obtained by issuing the query to different verticals. Arguello et al. proposed constructing smaller, representative corpora of vertical content rather than using the original vertical index. The representative corpora can be a sample from the vertical or a sample from surrogate corpora like Wikipedia.

Simple corpus-based features may include the number of relevant documents returned by verticals and ranking scores of the top ranked documents.

Another batch of features are those used for predicting query performance. One representative feature is Clarity proposed by Cronen-Townsend et al. [16]. Clarity is the relative entropy, or Kullback–Leibler divergence, between the language of the top ranked documents and the language of the collection. More specifically, Clarity of a query to a vertical v is calculated as follows:

$$Clarity(q, v) = \sum_{w} P(w|\theta_q) \log_2 \frac{P(w|\theta_q)}{P(w|\theta_{C_v})}. \tag{4.1}$$

Here w is a term from the vocabulary generated based on the document collection C_v of vertical v. $P(w|\theta_q)$ and $P(w|\theta_{C_v})$ are the query and collection language models, respectively. $P(w|\theta_q)$ is usually estimated by averaging the language models of the top retrieved documents of q. A low Clarity score usually means that random results are returned from the vertical, hence the query has low probability belonging to the vertical.

Another representative corpus-based feature is ReDDE, which is originally proposed by Si and Callan [51] for solving the resource selection problem. ReDDE is a resource-ranking algorithm, which estimates the distribution of relevant documents across the set of available verticals. It scores a target vertical based on the retrieval of an index that combines documents sampled from every target verticals. Given this retrieval, ReDDE accumulates a vertical's score from its document scores, taking into account the difference between the number of documents contained in the vertical and the number of documents sampled from the vertical. More specifically,

$$ReDDE(q, v) = |D_v| \sum_{d \in R} I(d \in S_v) P(q|\theta_d) P(d|S_v), \tag{4.2}$$

where $|D_v|$ is the number of documents in vertical v and S_v is the documents sampled from v. This feature is used by Arguello et al. [2] for vertical intent classification.

4.3.2.2 Query String-Based Features

Query string-based features aim to model the explicit expression of queries issued to search engines for seeking vertical contents. For each vertical, we can generate a list of handcrafted rules that can directly identify possible vertical intent of a query. For example, "[location] `weather` → weather" for weather intent identifies that each query comprised of a location name and the term "weather" has an explicit weather intent.

Tsur et al. [56] investigated the problem of detecting queries with a question intent. They called these queries as CQA-intent queries, since answers to them are typically found in community question answering (CQA) sites. As CQA-intent queries are usually long, they proposed to take the structure of queries into consideration for detecting CQA-intent queries. They extracted the following query string-based features: (1) the position of WH words in the query; (2) the number of tags the specific tags appear in the part-of-speech (POS) tagging result of the query.

4.3.2.3 Query Log-Based Features

Query log contains rich information about users' preferences on verticals. The vertical of a query can be estimated by evaluating the similarity between the query and all clicked documents within the vertical.

Arguello et al. [2] used the query likelihood given by a unigram language model constructed from the vertical's query log as a feature for classifying query vertical intent. Given a query q, the probability it belongs to a vertical v is defined by

$$QL(q, v) = \frac{P(q|\theta_v)}{\sum_{v' \in V} P(q|\theta_{v'})}, \tag{4.3}$$

where θ_v is vertical v's language model generated based on query log and V is a set of candidate verticals.

Kanhabua et al. [33] used query logs for detecting event-related queries (such as queries related to political elections, sport competitions, or natural disasters). More specifically, they used the normalized query volume aggregated across all users over time and the normalized click frequency for the query accumulated from all URLs and users as daily time series. In addition to these two data sources, they further used the temporal distribution of number of top-K search results retrieved from an external document collection as the third time series. For each time series, they extracted a list of features, including but not limited to: (1) Seasonality, which is a temporal pattern that indicates how periodic is an observed behavior over time. They used Holt–Winters adaptive exponential smoothing to decompose the time series and generated the seasonality component. Then they used trending scope and trending amplitude as features. (2) Autocorrelation, which is the cross correction of a signal with itself or the correlation between its own past and future values at different times. (3) Click entropy, which is proposed by Dou et al. [19], is used to model the temporal content dynamics. (4) Other features, including burstiness, kurtosis, and temporal KL-divergence. Information about more features can be found in [33].

Zhou et al. [67] used the query log-based features together with the query string-based features for vertical intent classification. They first identified vertical intent for a set of queries based on query string-based features we introduced in Sect. 4.3.2.2. For example, "Beijing weather" is predicted to have a weather intent because it contains the explicit keyword "weather." Queries containing "images," "picture," or "photo" are related to image vertical. Then, they classified URLs using the same rule-based method. For example, an URL containing a word "images" will be classified into image vertical. All clicked URLs made on a vertical query are also assumed to belong the same vertical. Finally, for a given query q and a vertical v, they calculated the fraction of clicks that linked to pages in the vertical, compared to the number of total clicks for the query, and used a threshold to identify whether q is related to vertical v.

4.3.2.4 Search Results-Based Features

In addition to the corpus-based features, which mainly rely on the documents returned from the verticals or representative corpora of verticals, we can also develop features based on characteristic of search results returned by the general Web search.

The first type of information we can utilize is the statistics of websites within the results. If the results of a query contain many websites, which are typical websites of a vertical, the query is possibly relevant to the vertical.

The second type of information is the keywords or phrases contained in the snippets or the content of the search results. For example, if the snippets of search results of a query contain the keywords "film" or "movie" frequently, the query may have a movie intent.

4.3.2.5 Vertical Intent Classification Models

Similar to topical intent classification, most existing vertical intent classification (or vertical selection) approaches [2, 33, 56] are based on supervised learning-based algorithms, such as Logistic Regression, SVM, Random Forest, and Gradient Boosted Decision Tress (GBDT). Studies have shown that when trained using a large set of labeled data, a machine learned vertical selection model outperforms baselines that require no training data [3].

One problem of the supervised classifiers is that whenever a new vertical is introduced, a costly new set of editorial data must be gathered. To solve this problem, Arguello et al. [3] proposed methods for reusing training data from a set of existing verticals to learn a predictive model for a new vertical. Their experiments showed the need to focus on different types of features when maximizing portability (the ability for a single model to make accurate predictions across multiple verticals) than when maximizing adaptability (the ability for a single model to make accurate predictions for a specific vertical). Hu et al. [25] also revealed that it is a big challenge to create training data for statistical machine learning-based query vertical classification approaches. They proposed a general methodology to discover large quantities of intent concepts by leveraging Wikipedia, which required very little human effort. Within this framework, each intent domain is represented as a set of Wikipedia articles and categories, and the intent of a query is identified through mapping the query into the Wikipedia representation space. Based on their study on three different vertical classification tasks, i.e., travel, job, and person name, this approach achieved much better coverage than previous approaches to classify queries in an intent domain even through the number of seed intent examples is very small. Li et al. [35, 36] used click graphs to automatically infer class memberships of unlabeled queries from those of labeled ones based on the co-click behaviors of users. They then used these automatically labeled queries to train content-based query classification models using query terms as features. Their experimental results on product intent classification and job intent classification indicated that by using a large amount of training queries obtained in this way, classifiers using only query term or lexical features (without the use of features from search results) can outperform those using augmented features from external knowledge.

4.4 Query Intent Mining

A large percentage of queries issued to search engines are broad or ambiguous [19, 20, 28, 45, 52]. By submitting one query, users may have different intents or information need. For an ambiguous query, users may seek for different interpretations; whereas for a query on a broad topic, users may be interested in different subtopics. For example, by issuing the ambiguous query `[apple]`, one user might be searching for information about the IT company Apple, whereas another user might be looking for information about apple fruit. By issuing a broad query `[harry potter]`, a user may want to seek content covering various aspects, such as `[harry potter movie]`, `[harry potter book]`, or `[harry potter characters]` within this broad topic. Without accurately understanding users' underlying intents of a query, search engines may fail to return enough results that can cover major intents in the top ranks, hence may affect search experience of some users. So it is critical to mine underlying intents of a query.

Query intent mining, which is called **subtopics mining** sometimes, is an essential step to search result diversification, which aims to solve the problem of query ambiguity. Search result diversification aims to return diverse search results that cover as many user intents as possible. It has received a lot of attention in recent years. Many search result diversification algorithms [1, 12, 13, 17, 21, 43, 45, 46, 49, 63, 68] have been developed to improve search result diversity. A common characteristic of most existing explicit diversification algorithms is that they assume the existence of a flat list of independent subtopics [17, 21, 49]. Table 4.4 shows the manually created subtopics for query "defender" (topic number 20) in TREC 2009 [14]. There are five distinct subtopics for the query. For subtopics s_1, s_3, and s_5, users are all looking for different information about a software "Windows Defender". For subtopic s_2, users are interested in general information about a brand of car "Land Rover Defender." For subtopic s_4, users are finding specific information about playing a "Defender arcade game" online.

The Subtopic Mining subtask in NTCIR-9 Intent task [54] and NTCIR-10 Intent-2 task [48] aimed to have an evaluation of intent mining approaches. In the Subtopic Mining subtask, systems were required to return a ranked list of subtopic strings in response to a given query. A subtopic could be a specific interpretation of an

Table 4.4 Subtopics of query "defender"

No.	Subtopic description
s_1	I'm looking for the homepage of Windows Defender, an antispyware program
s_2	Find information on the Land Rover Defender sport-utility vehicle
s_3	I want to go to the homepage for Defender Marine Supplies
s_4	I'm looking for information on Defender, an arcade game by Williams. Is it possible to play it online?
s_5	I'd like to find user reports about Windows Defender, particularly problems with the software

ambiguous query (e.g., "Microsoft windows" or "house windows" in response to "windows") or an aspect of a faceted query (e.g., "windows 7 update" in response to "windows 7"). The subtopics collected from participants were pooled and manually assessed. The Subtopic Mining subtask received 42 Chinese runs and 14 Japanese runs in NTCIR-9. INTENT-2 attracted participating teams from China, France, Japan, and South Korea—12 teams for Subtopic Mining, and it received 34 English runs, 23 Chinese runs, and 14 Japanese runs. More details about these evaluation tasks can be found in [54] and [48]. A similar task is the I-Mine task [38, 61] in NTCIR-11 and NTCIR-12.

In the remaining part of this section, we will briefly introduce existing approaches for mining query intent or subtopics.

4.4.1 Mining Intent from Query Logs

Query log data contain much useful information about user intents, as queries are directly issued by real-world users. When a user issues the query that may be ambiguous or underspecified and does not get expected results, she often refines the query and resubmits a new query to search engines. So by analyzing the query strings, reformulation, follow-up, and co-click behavior in query logs, it is able to identify user intents.

4.4.1.1 Mining Intent from Query Strings and Sessions

The most simple way to mine intents for a query is directly retrieving longer queries started or ended with the original query. A longer query containing the original query usually stands for a narrower intent, hence it is reasonable to directly take the longer queries as subintents. As there might be a large number of queries containing a short query, usually only the top n extended queries with the highest frequencies are selected.

Strohmaier et al. [55] obtained similar queries from search sessions, filtered out noisy queries using click-through data, and then grouped the remaining queries based on random walk similarity. They also estimated the popularity of each intent based on the number of observations in the query logs.

4.4.1.2 Mining Intent Based on Reformulation Behavior

Radlinski and Dumais [43] proposed to use the reformulation behavior of users within query logs to find likely user intents. Dou et al.[21] refined this method and used it to generate subtopics from query log for search result diversification.

Suppose for each query q_i, n_i is the number of times the query was issued. For a pair of queries (q_i, q_j), let n_{ij} be the number of times q_i was followed by q_j. The empirical probability of q_i being followed by q_j can be defined as follows:

$$p_{ij} = \frac{n_{ij}}{n_i}. \tag{4.4}$$

The problem of directly using the empirical follow-up probability p_{ij} is that follow-up queries are usually dominated by top user intents. For example, top three follow-up queries for query "defender" are "windows defender download," "Microsoft defender," and "windows defender" in a real search engine. These queries are actually talking about the same intent related to "windows defender." To retrieve more diverse intents, an MMR-like [12] measure can be used to greedily select the set of queries that are related to the given query yet different from each other.

Suppose $R(q_i)$ is the set of queries (subtopics) already selected, the next best query, namely q^n, is selected by:

$$q^n = \arg\max_{q_j} \left[\lambda \cdot p_{ij} - (1 - \lambda) \cdot \max_{q_k \in R(q_i)} sim(q_j, q_k) \right], \tag{4.5}$$

where λ is a parameter to control the similarity between returned intents (queries). $sim(q_j, q_k)$ is the similarity between two queries q_j and q_k.

We assume that the two queries q_j and q_k are similar if:

- q_j and q_k are frequently co-issued in the same query sessions. The probability of two queries being issued together in the same query sessions can be evaluated by the measurement $p^*_{jk} = \sqrt{p_{jk} p_{kj}}$ proposed by Radlinski and Dumais [43]. A high p^*_{jk} value means that q_j and q_k are frequently issued in the same sessions.
- The results by searching q_j and q_k are similar. Suppose $Docs(q_j)$ and $Docs(q_k)$ are top ten search results returned for query q_j and q_k. Dou et al. [21] used $\frac{|Docs(q_j) \cap Docs(q_k)|}{|Docs(q_j) \cup Docs(q_k)|}$ to evaluate the result similarity of these two queries.
- The words contained in q_j and q_k are similar. Dou et al. used $\frac{|q_j \cap q_k|}{|q_j \cup q_k|}$ to measure the text similarity between these two queries.

Dou et al. [21] then used a linear combination of these factors as follows and used it in Eq. (4.5) to rank queries as subtopics:

$$sim(q_j, q_k) = \frac{1}{3} \left\{ p^*_{jk} + \frac{|Docs(q_j) \cap Docs(q_k)|}{|Docs(q_j) \cup Docs(q_k)|} + \frac{|q_j \cap q_k|}{|q_j \cup q_k|} \right\}. \tag{4.6}$$

Example subtopics mined from query logs for the query "defender" are shown in Table 4.5.

Table 4.5 Subtopics of query "defender" mined from query logs

Subtopic	Rank	Subtopic	Rank
Windows defender download	1	Defender marine supply	6
Defender arcade game	2	Install Microsoft defender	7
Defender antivirus	3	Defender for XP	8
Land rover defender	4	Microsoft defender review	9
Free windows defender beta	5	Defender pro	10

4.4.1.3 Mining Intent from Click Graph

Radlinski et al. [44] proposed to combine reformulation and click information within query logs to find likely user intents.

To mine query intent, they first identified a set of possibly related queries to a query q by retrieving the k most frequent valid reformulations of q, and the k most frequent valid reformulations of these direct reformulations. Here "valid" means that the formulation is made by enough users (e.g., at least 2 users in [44]), and the probability of this formulation made among all formulations is larger than a threshold (Radlinski et al. used 0.001 as the threshold in [44]). They then removed queries less related to the original query by using a two-step random walk on the bipartite query-document click graph. Only those queries that have similar clicks with the original queries can be kept. Last, the left queries are clustered based on their similarities within the click graph based on random walk.

Hu et al. [27] employed both expanded queries and click graph to mine query intents. The entire solution is similar to Radlinski et al. [44]. They assumed that documents clicked in a specific search are likely to represent the same underlying intent. They grouped the URLs associated with a query and its expanded queries into clusters and then used expanded queries associated with the clusters to describe the intents.

4.4.2 Mining Intent from Search Results

A typical way for mining intent from search results is search result clustering [59, 65]. Zeng et al. [65] reformalized the search result clustering problem as a supervised salient phrase ranking problem. Given a query, they first extracted and ranked salient phrases as candidate cluster names, based on a regression model learned from human-labeled training data. The documents are assigned to relevant salient phrases to form candidate clusters, and the final clusters are generated by merging these candidate clusters.

Dou et al. [21] treated each cluster as an implicit subtopic/intent. They assumed that a cluster (subtopic), denoted by $cluster_1$, is more important than another cluster, denoted by $cluster_2$, if: (1) $cluster_1$ is ranked higher than $cluster_2$ in terms of salient phrases; and (2) the best document within the cluster $cluster_1$ is ranked higher than that in $cluster_2$. They then employed the following equation based on the above two assumptions to evaluate the importance of a cluster subtopic:

$$w(q, c) = 0.5 \times \frac{K - \text{clstRank}_c + 1}{K} + 0.5 \times \frac{1}{\text{bestDocRank}_c}, \tag{4.7}$$

where clstRank_c is the rank of the cluster among all clusters, and bestDocRank_c is the highest rank of the documents within the cluster, i.e., $\text{bestDocRank}_c = \min_{d \in c} \text{rank}_d$. They used the same settings $N = 200$ and $K = 10$ as those in [65].

Wang et al. [57] used surrounding text of query terms in top retrieved documents to mine intent. They first extracted text fragments containing query terms from different parts of documents. Then they grouped similar text fragments into clusters and generated a readable subtopic for each cluster. Based on the cluster and the language model trained from a query log, they calculated three features and combined them into a relevance score for each subtopic. Subtopics were finally ranked by balancing relevance and novelty. Their evaluation experiments with the NTCIR-9 INTENT Chinese Subtopic Mining test collection show that the proposed method significantly outperformed a query log-based method proposed by Radlinski et al. [44] and a search result clustering-based method proposed by Zeng et al. [65] in terms of the official evaluation metrics used at the NTCIR-9 INTENT task. Moreover, the generated subtopics were significantly more readable than those generated by the search result clustering method.

4.4.3 Mining Intent from Anchor Texts

Anchor texts created by Web designers provide meaningful descriptions of destination documents. They are usually short and descriptive, which share the similar characteristics with Web queries. Given a query, anchor texts that contain the query terms usually convey the information about the query intents, hence it is reasonable to use these kinds of related anchor texts as query intents or subtopics.

Dou et al. [21] mined query intent from anchor text for search result diversification. For a given query q, they retrieved all anchor texts containing all query terms of q, weighted them, and selected the most important ones as subtopics. They assumed that the importance of an anchor text is usually proportional to its popularity on the Web, i.e., how many times it is used in Web sites or pages. However, a shorter anchor text usually matches the query better than a longer anchor text. The subtopic of the longer anchor text may be overspecified or drifted from the original query. Based on these observations, they design the following ranking function to evaluate

Table 4.6 Subtopics of query "defender" mined from anchor text in ClueWeb09 document corpus

Subtopic	Rank	Subtopic	Rank
Castle defender	1	Reputation defender	6
Public defender	2	Star defender	7
Cosmic defender	3	Chicago defender	8
Windows defender	4	Base defender	9
Brewery defender	5	Doodle defender	10

the importance of an anchor text c:

$$\begin{aligned} f(q, c) &= \text{freq}(c) * \text{rel}(q, c) \\ &= \left[\text{nsite}_c + \log(\text{npage}_c - \text{nsite}_c + 1)\right] * \frac{1 + \text{len}(q)}{\text{len}(c)}. \end{aligned} \tag{4.8}$$

The first term $\text{freq}(c) = \text{nsite}_c + \log(\text{npage}_c - \text{nsite}_c + 1)$ evaluates the popularity of anchor text c, in which npage_c denotes the number of source pages that contain the anchor text c, and nsite_c denotes the number of unique source sites of these links. As it is easy to create a large number of source pages within the same source site to boost the anchor text, in the above equation, each source site just counts once. Additional pages containing the anchor text (totally $\text{npage}_c - \text{nsite}_c$ pages) from these sites are assigned lower weights by discounting their votes using the log function. Obviously an anchor text used by a larger number of different websites will get a high value of $\text{freq}(c)$.

The second term $\text{rel}(q, c) = \frac{1+\text{len}(q)}{\text{len}(c)}$ punishes the anchor texts that contain too many words. Note that $\text{len}(q)$ is the count of query terms, and $\text{len}(c)$ is the number of terms contained in c. For the query q, an anchor text $q + t_1$ with an additional term t_1 gets as high $\text{rel}(q, c)$ as one, because it is a perfect subtopic of the query; whereas, another one $q + t_1 + t_2$ containing two additional terms gets lower $\text{rel}(q, c)$.

Table 4.6 shows the top 10 anchor texts with their weights for the query "defender" mined from the ClueWeb09 [15] collection.

4.4.4 Mining Intent from Query Suggestions

Another data source for mining intents is query suggestions. Query suggests are widely used resources for mining intent. Some search result diversification approaches directly utilized query suggestions as query intents or subtopics [17, 21, 49]. Search engines generate query suggests to users, to let them simply navigate to a better query when they are not satisfied by the current results. The query suggestions can be directly extracted from the search result page, and this is the reason why they are widely used in academic when there is no query log data.

4.4.5 Mining Complex Intents

All the above intent mining approaches assume the existence of a flat list of independent subtopics. However, it is hard to say these subtopics could reflect the complex information needs of users. Furthermore, most intent lists are mined from a single data source, whereas different data sources may help reflect the uncertainty of a query from different perspectives. For example, query logs reflect the popular requirements of real-world users, whereas anchor texts give an overview of the possible meanings of a query that is less biased by users and search engines. At the same time, the sole use of one data source or one mining algorithm may fail to satisfy the various requirements of different users, for example, when they are used for search result diversification [19]. Query logs are not available for new queries, and they have bias toward background rankings. Anchor texts can conquer these shortcomings instead. Query logs and anchor texts are applicable for short and popular queries; whereas subtopics mined from search results may work for both popular and tail queries.

As different types of subtopics are complimentary to each other, combining them together can potentially help the applications (such as search result diversity). Dou et al. [21] proposed a general framework of diversifying search results based on multiple dimensions of subtopics.

Hu et al. [26] revealed that user intents covered by a query can be hierarchical. They leveraged hierarchical intents and proposed hierarchical diversification models to promote search result diversification. Similar to previous works [17, 49], they used query suggestions extracted from Google search engine as subtopics. For each query, we collected its query suggestions from Google as the first-level subtopics. To generate subtopic hierarchy, they further issued the first-level subtopics as queries to Google and retrieved their query suggestions as the second-level subtopics. Finally, they collect 1696 first-level subtopics and 10,527 second-level subtopics for 194 TREC Web track queries. They assumed a uniform probability distribution for all the first-level subtopics and assumed a uniform probability distribution for the second-level subtopics with respect to their parent subtopics. Experimental results showed that using the hierarchical intent structures outperformed the use of flat intent list.

Wang et al. [58] also investigated the problem of hierarchical intents. They modeled user intents as intent hierarchies and used the intent hierarchies for evaluating search result diversity. They proposed several diversity measures based on intent hierarchies and demonstrated that in some cases, the new measures outperformed the original corresponding measures.

4.5 Other Kinds of Intent Classification

In addition to the general intent classification task, researchers also investigated solutions for classifying specific intents, such as temporal intent [33, 66] and geographic intent [62].

4.5.1 *Temporal Intent Classification*

Kanhabua et al. [33] studied the problem of detecting event-related queries. They used seasonality, autocorrelation, click entropy, kurtosis, and many other features to model the patterns of the time series extracted from query logs and document corpus. Differently, Zhao et al. [66] explored the usage of time-series data derived from Wikipedia page views, a freely available data source, for temporal intent disambiguation. They also used seasonality, autocorrelation, and other time-series-based features. Hasanuzzaman et al. [23] used 11 independent features extracted from the temporal information contained in the query string, its issuing date, and the extra data collected.

4.5.2 *Geographic Intent Classification*

Yi et al. [62] addressed the geo intent detection problem. They created a city language model, which is a probabilistic representation of the language surrounding the mention of a city in Web queries. They used several features derived from these language models to identify users' implicit geo intent or predict cities for queries that contain location-related entities.

References

1. Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Proceedings of the Second International Conference on Web Search and Data Mining9*, pages 5–14, 2009.
2. Jaime Arguello, Fernando Diaz, Jamie Callan, and Jean-François Crespo. Sources of evidence for vertical selection. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–322, 2009.
3. Jaime Arguello, Fernando Diaz, and Jean-François Paiement. Vertical selection in the presence of unlabeled verticals. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 691–698, 2010.
4. Ricardo A. Baeza-Yates, Liliana Calderón-Benavides, and Cristina N. González-Caro. The intention behind web queries. In *Proceedings of the 13th International Conference on String Processing and Information Retrieval*, pages 98–109, 2006.

5. Steven M. Beitzel, Eric C. Jensen, Ophir Frieder, David D. Lewis, Abdur Chowdhury, and Aleksander Kolcz. Improving automatic query classification via semi-supervised learning. In *Proceedings of the 5th IEEE International Conference on Data Mining*, pages 42–49, 2005.
6. Steven M. Beitzel, Eric C. Jensen, Abdur Chowdhury, and Ophir Frieder. Varying approaches to topical web query classification. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 783–784, 2007.
7. Jiang Bian, Tie-Yan Liu, Tao Qin, and Hongyuan Zha. Ranking with query-dependent loss for web search. In *Proceedings of the Third International Conference on Web Search and Data Mining*, pages 141–150, 2010.
8. David J. Brenes and Daniel Gayo-Avello. Automatic detection of navigational queries according to behavioural characteristics. In *Proceedings of the LWA 2008 - Workshop-Woche: Lernen, Wissen & Adaptivität*, pages 41–48, 2008.
9. David J. Brenes, Daniel Gayo-Avello, and Kilian Pérez-González. Survey and evaluation of query intent detection methods. In *Proceedings of the 2009 workshop on Web Search Click Data*, pages 1–7, 2009.
10. Andrei Z. Broder. A taxonomy of web search. *SIGIR Forum*, 36 (2): 3–10, 2002.
11. Andrei Z. Broder, Marcus Fontoura, Evgeniy Gabrilovich, Amruta Joshi, Vanja Josifovski, and Tong Zhang. Robust classification of rare queries using web knowledge. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 231–238, 2007.
12. Jaime G. Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336, 1998.
13. Harr Chen and David R. Karger. Less is more: probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 429–436, 2006.
14. Charles L. A. Clarke, Nick Craswell, and Ian Soboroff. Overview of the TREC 2009 web track. In *Proceedings of The Eighteenth Text REtrieval Conference*, volume 500–278, 2009.
15. ClueWeb09. The clueweb09 dataset. http://boston.lti.cs.cmu.edu/Data/clueweb09/.
16. Stephen Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 299–306, 2002.
17. Van Dang and W. Bruce Croft. Diversity by proportionality: an election-based approach to search result diversification. In *Proceedings of the 35th International ACM SIGIR conference on research and development in Information Retrieval*, pages 65–74, 2012.
18. Vincenzo Deufemia, Massimiliano Giordano, Giuseppe Polese, and Luigi Marco Simonetti. Exploiting interaction features in user intent understanding. In *Proceedings of the 15th Asia-Pacific Web Conference*, pages 506–517, 2013.
19. Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. A large-scale evaluation and analysis of personalized search strategies. In *Proceedings of the 16th International Conference on World Wide Web*, pages 581–590, 2007.
20. Zhicheng Dou, Ruihua Song, Ji-Rong Wen, and Xiaojie Yuan. Evaluating the effectiveness of personalized web search. *IEEE Trans. Knowl. Data Eng.*, 21 (8): 1178–1190, 2009.
21. Zhicheng Dou, Sha Hu, Kun Chen, Ruihua Song, and Ji-Rong Wen. Multi-dimensional search result diversification. In *Proceedings of the Forth International Conference on Web Search and Data Mining*, pages 475–484, 2011.
22. Qi Guo and Eugene Agichtein. Exploring mouse movements for inferring query intent. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 707–708, 2008.
23. Mohammed Hasanuzzaman, Sriparna Saha, Gaël Dias, and Stéphane Ferrari. Understanding temporal query intent. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 823–826, 2015.

24. Mauro Rojas Herrera, Edleno Silva de Moura, Marco Cristo, Thomaz Philippe C. Silva, and Altigran Soares da Silva. Exploring features for the automatic identification of user goals in web search. *Inf. Process. Manage.*, 46 (2): 131–142, 2010.
25. Jian Hu, Gang Wang, Frederick H. Lochovsky, Jian-Tao Sun, and Zheng Chen. Understanding user's query intent with Wikipedia. In *Proceedings of the 18th International Conference on World Wide Web*, pages 471–480, 2009.
26. Sha Hu, Zhicheng Dou, Xiao-Jie Wang, Tetsuya Sakai, and Ji-Rong Wen. Search result diversification based on hierarchical intents. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*, pages 63–72, 2015.
27. Yunhua Hu, Ya-nan Qian, Hang Li, Daxin Jiang, Jian Pei, and Qinghua Zheng. Mining query subtopics from search log data. In *Proceedings of the 35th International ACM SIGIR conference on research and development in Information Retrieval*, pages 305–314, 2012.
28. Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Inf. Process. Manag.*, 36 (2): 207–227, 2000.
29. Bernard J. Jansen, Danielle L. Booth, and Amanda Spink. Determining the user intent of web search engine queries. In *Proceedings of the 16th International Conference on World Wide Web*, pages 1149–1150, 2007.
30. Bernard J. Jansen, Danielle L. Booth, and Amanda Spink. Determining the informational, navigational, and transactional intent of web queries. *Inf. Process. Manag.*, 44 (3): 1251–1266, 2008.
31. In-Ho Kang. Transactional query identification in web search. In *Proceedings of the Second Asia Information Retrieval Symposium*, pages 221–232, 2005.
32. In-Ho Kang and Gil-Chang Kim. Query type classification for web document retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 64–71, 2003.
33. Nattiya Kanhabua, Tu Ngoc Nguyen, and Wolfgang Nejdl. Learning to detect event-related queries for web search. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1339–1344, 2015.
34. Uichin Lee, Zhenyu Liu, and Junghoo Cho. Automatic identification of user goals in web search. In *Proceedings of the 14th international conference on World Wide Web*, pages 391–400, 2005.
35. Xiao Li, Ye-Yi Wang, and Alex Acero. Learning query intent from regularized click graphs. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 339–346, 2008.
36. Xiao Li, Ye-Yi Wang, Dou Shen, and Alex Acero. Learning with click graph for query intent classification. *ACM Trans. Inf. Syst.*, 28 (3): 12:1–12:20, 2010.
37. Yiqun Liu, Min Zhang, Liyun Ru, and Shaoping Ma. Automatic query type identification based on click through information. In *Proceedings of the Third Asia Information Retrieval Symposium*, pages 593–600, 2006.
38. Yiqun Liu, Ruihua Song, Min Zhang, Zhicheng Dou, Takehiro Yamamoto, Makoto P. Kato, Hiroaki Ohshima, and Ke Zhou. Overview of the NTCIR-11 imine task. In *Proceedings of the 11th NTCIR Conference on Evaluation of Information Access Technologies*, 2014.
39. Yuchen Liu, Xiaochuan Ni, Jian-Tao Sun, and Zheng Chen. Unsupervised transactional query classification based on webpage form understanding. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management*, pages 57–66, 2011.
40. Yumao Lu, Fuchun Peng, Xin Li, and Nawaaz Ahmed. Coupling feature selection and machine learning methods for navigational query identification. In *Proceedings of the 2006 ACM CIKM International Conference on Information and Knowledge Management*, pages 682–689, 2006.
41. Marcelo Mendoza and Juan Zamora. Identifying the intent of a user query using support vector machines. In *Proceedings of the 16th International Symposium on String Processing and Information Retrieval*, pages 131–142, 2009.
42. David Nettleton, Liliana Calderón-benavides, and Ricardo Baeza-yates. Analysis of web search engine query sessions. In *Proceedings of WebKDD 2006: KDD Workshop on Web Mining and Web Usage Analysis, in conjunction with the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

43. Filip Radlinski and Susan T. Dumais. Improving personalized web search using result diversification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 691–692, 2006.
44. Filip Radlinski, Martin Szummer, and Nick Craswell. Inferring query intent from reformulations and clicks. In *Proceedings of the 19th International Conference on World Wide Web*, pages 1171–1172, 2010.
45. Davood Rafiei, Krishna Bharat, and Anand Shukla. Diversifying web search results. In *Proceedings of the 19th International Conference on World Wide Web*, pages 781–790, 2010.
46. Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. Toward whole-session relevance: exploring intrinsic diversity in web search. In *Proceedings of the 36th International ACM SIGIR conference on research and development in Information Retrieval*, pages 463–472, 2013.
47. Daniel E. Rose and Danny Levinson. Understanding user goals in web search. In *Proceedings of the 13th international conference on World Wide Web*, pages 13–19, 2004.
48. Tetsuya Sakai, Zhicheng Dou, Takehiro Yamamoto, Yiqun Liu, Min Zhang, and Ruihua Song. Overview of the NTCIR-10 INTENT-2 task. In *Proceedings of the 10th NTCIR Conference on Evaluation of Information Access Technologies*, 2013.
49. Rodrygo L. T. Santos, Jie Peng, Craig Macdonald, and Iadh Ounis. Explicit search result diversification through sub-queries. In *Proceedings of the 32nd European Conference on IR Research*, pages 87–99, 2010.
50. Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Building bridges for web query classification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 131–138, 2006.
51. Luo Si and James P. Callan. Relevant document distribution estimation method for resource selection. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 298–305, 2003.
52. Craig Silverstein, Monika Rauch Henzinger, Hannes Marais, and Michael Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33 (1): 6–12, 1999.
53. Ruihua Song, Zhenxiao Luo, Jian-Yun Nie, Yong Yu, and Hsiao-Wuen Hon. Identification of ambiguous queries in web search. *Inf. Process. Manag.*, 45 (2): 216–229, 2009.
54. Ruihua Song, Min Zhang, Tetsuya Sakai, Makoto P. Kato, Yiqun Liu, Miho Sugimoto, Qinglei Wang, and Naoki Orii. Overview of the NTCIR-9 INTENT task. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, 2011.
55. Markus Strohmaier, Mark Kröll, and Christian Körner. Intentional query suggestion: making user goals more explicit during search. In *Proceedings of the 2009 workshop on Web Search Click Data*, pages 68–74, 2009.
56. Gilad Tsur, Yuval Pinter, Idan Szpektor, and David Carmel. Identifying web queries with question intent. In *Proceedings of the 25th International Conference on World Wide Web*, pages 783–793, 2016.
57. Qinglei Wang, Ya-nan Qian, Ruihua Song, Zhicheng Dou, Fan Zhang, Tetsuya Sakai, and Qinghua Zheng. Mining subtopics from text fragments for a web query. *Inf. Retr.*, 16 (4): 484–503, 2013.
58. Xiao-Jie Wang, Ji-Rong Wen, Zhicheng Dou, Tetsuya Sakai, and Rui Zhang. Search result diversity evaluation based on intent hierarchies. *IEEE Trans. Knowl. Data Eng.*, 30 (1): 156–169, 2018.
59. Xuanhui Wang and ChengXiang Zhai. Learn from web search logs to organize search results. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 87–94, 2007.
60. Yu Wang and Eugene Agichtein. Query ambiguity revisited: Clickthrough measures for distinguishing informational and ambiguous queries. In *Proceedings of the Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*, pages 361–364, 2010.

61. Takehiro Yamamoto, Yiqun Liu, Min Zhang, Zhicheng Dou, Ke Zhou, Ilya Markov, Makoto P. Kato, Hiroaki Ohshima, and Sumio Fujita. Overview of the NTCIR-12 imine-2 task. In *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*, 2016.
62. Xing Yi, Hema Raghavan, and Chris Leggetter. Discovering users' specific geo intention in web search. In *Proceedings of the 18th International Conference on World Wide Web*, pages 481–490, 2009.
63. Yisong Yue and Thorsten Joachims. Predicting diverse subsets using structural SVMs. In *Proceedings of the Twenty-Fifth International Conference on Machine Learning*, pages 1224–1231, 2008.
64. Juan Zamora, Marcelo Mendoza, and Héctor Allende. Query intent detection based on query log mining. *J. Web Eng.*, 13 (1&2): 24–52, 2014.
65. Hua-Jun Zeng, Qi-Cai He, Zheng Chen, Wei-Ying Ma, and Jinwen Ma. Learning to cluster web search results. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 210–217, 2004.
66. Yue Zhao and Claudia Hauff. Temporal query intent disambiguation using time-series data. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 1017–1020, 2016.
67. Ke Zhou, Ronan Cummins, Martin Halvey, Mounia Lalmas, and Joemon M. Jose. Assessing and predicting vertical intent for web queries. In *Proceedings of the 34th European Conference on IR Research*, pages 499–502, 2012.
68. Xiaojin Zhu, Andrew B. Goldberg, Jurgen Van Gael, and David Andrzejewski. Improving diversity in ranking using absorbing random walks. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 97–104, 2007.