A SURVEY: FROM IR TO WEB IR

Ping-I Chen

Digital Education Institute, Institute for Information Industry, Taipei, Taiwan, ROC
be@iii.org.tw

ABSTRACT

As Internet use continues to grow, users become more and more reliant on Web information instead of documents or databases in local sites. Information retrieval techniques also need to go with the flow and try to create more accurate and efficient ways to deal with information retrieval. We will introduce several important algorithms and research articles that have been used in the information retrieval (IR) area for several years to summarize previous research and to see whether or not these algorithms can be used in the Internet environment of the future. Mining on the Web to provide the immediate information is difficult because of the heterogeneous source providers, knowledge domains, and different languages. By using a Normalized Google Distance (NGD) algorithm, which uses Google as a semantic corpus, we can provide a new aspect for IR research and extract the most important keywords or keyword sequences for advanced knowledge discovery. We can understand the evolution of the technique from IR to Web IR. Also, we will provide some new ideas for applications of IR research that will require no training process, and all the executions are on-line and in real time.

Keywords: Information retrieval, Web IR, NGD, machine learning.

1. Introduction

In the past, if a company wanted to collect information to aid in business decision making, they would send many employees all over the world to communicate with the locals and then write a report on their findings to send back to the company. However, the Internet now has become an indispensable part of our daily life, and many new applications, such as powerful search engines, online virtual communities, and others, make our work style, knowledge transformation, and personal interactions more convenient. We can instantly gather all kinds of information from around the world. Thus, the new problem has become this: How can we filter out the irrelevant information and find the things we really need or want to know?

For Web IR, the search engine will produce many related query results, but the users may want to know only the highest relevant information. Likewise, browsing and searching behavior on the Internet is more diversified. For instance, we may be reading an article on sports news and then decide we want to read some Economic research papers and want to be able to change direction immediately. For this reason, a new technique needs to be invented that can handle the problem of cross knowledge domain information retrieval. The instantaneous provision of information on the Internet is also another chief user concern. People do not have enough time to train on the statistical model or to cluster the related documents and use them to find the information they are seeking. Cilibrasi et al. (2007) proposed a new idea called the Normalized Google Distance (NGD) algorithm, which uses each keyword to conduct a search in Google to obtain the number of search results, and then uses that number to calculate the relationship between those keywords. They treated the Google search engine as the biggest corpus in the world, and one can find all the semantic relations from it without collecting them or constructing a training database and learning procedure. We believed that by using this kind of algorithm, we could achieve the goal of “What you see is what you get.” We can browse a Webpage, and the system will then automatically calculate the most important keywords and provide the information to locate more advanced reading about them. This can enable us to read an article that is not written in our first language or is not even in our major field.

Additionally, we can use the same system for document clustering, keyword expansion, knowledge mapping and other functions. Therefore, all the applications in the IR or the Web IR can become on-line real-time executions without the need to undergo any training, and the only
limitation will be the speed of Internet. Some people may question that asking, Why not the computing power? We believe that cloud computing or grid computing techniques can reduce the execution time of the Web IR process. But if the Internet is out of service and we cannot connect to Google’s search engine, then the NGD algorithm would be useless.

In this paper, we have reviewed many research articles and introduced some very important algorithms and applications related to information retrieval. These techniques are represented as figure 1. At the bottom of figure 1 we see icons for the various sources that provide information to Internet users in either textual or graphic formats. Numerous research articles focus on documents that are just for a single user only and provide only custom-designed information. The databases for these represent document sets that are solely owned by a company or by small groups, and this data is always highly correlated to specific knowledge domains. It can provide collaborative recommendations to the user so that one can obtain much information on a specific topic from the colleague of the company. Users can browse Internet Webpages that present information on personal behavior or other specific interests. They can also find customized information such as documents. But the data and the knowledge domains available are highly divergent. The fourth source of information provided on the Internet comes from various virtual communities available on the Web. The advantage of community-based information is that they all share a similar database. Nowadays, many virtual community forums are available on the Internet that enable users who have similar interests to join in a specific ongoing discussion or introduce new products. For example, the Mobile01 (http://www.mobile01.com/) is a famous forum in Taiwan, and it focuses on 3C products. The knowledge available through the community databases is always less than what is available on the Worldwide Web, and its content is provided only by members of that community.

These four source providers (documents, Webpages, databases, and communities) have different features, but the techniques used to extract keywords using algorithms (see middle of Fig. 1) and provide advanced applications (see upper part of Fig. 1) are nearly the same. The automatic term recognition and term weighting process will be introduced in section 2 of this paper. These processes can provide the user term-level information, such as a keyword suggestion, and also provide feedback to the source provider to improve reading or browsing efficiency. Two other applications will be discussed in sections 3 and 4; they are traditional applications and future-generation applications, respectively. The application level provides the user with advanced concept-level information by using the knowledge map. The newest applications in this area are joint inference and sentiment analysis, which are based on techniques used in traditional applications and provide a much higher level of accurate information for decision making.
2. Automatic Term Recognition (ATR) and Term Weighting

Automatic term recognition (ATR), in particular, is much needed because it is the starting point of information retrieval technology. The importance of words in the document are analyzed using frequency, location, and length of words to identify significant words. Automatic term recognition can be divided into three main categories: statistics approach, machine-learning approach, and semantic similarity (see Fig. 2).
1) Statistics Approach

This method is simple and does not require training data. The statistics information of the words is used to identify the keywords in the document. The most famous and frequently used is frequency-based TF-IDF algorithm. The TF-IDF is often used in information retrieval and text mining. Importance increases proportionally to the number of times a word appears in the document, but it is offset by the frequency of the word in the corpus. The TF-IDF weighting schemes are often used by search engines to score and re-rank a document’s relevance based on user queries.

Kageura et al. (1996) summarized some previous research and proposed several new ideas for how to determine the importance of keywords. For individual documents, we need to assume that the frequency with which a word appears in a document is likely to be an index term. Also, if it is a word that appears only in a limited number of documents, we can use those specific words as an index term for these documents. We also need to consider whether the keyword will appear more frequently in a document or show a specific distributional characteristic in the database.

One author (Chien, 1997; Chien, 1999) reported on the use of the PAT tree algorithm to estimate complete lexical patterns (CLPs). This method treats the entire text as an array of characters and uses a set of semi-infinite strings (sistrings) to construct the PAT tree structure. It was learned that most of the CLPs have strong associations between their composed and overlapped substrings so that one can measure the overlap of each of the two sistrings to learn how they are connected. This method works well in extracting Chinese names.

If a user is browsing a Webpage to search for unknown information by using a search engine, the user must always highlight the keywords and copy them to the search engine. Thus, a keyword extraction algorithm without a corpus is necessary in order to search effectively this way. Also without the limitation of the corpus, the user has access to all kinds of information regardless of which knowledge domain they are reading or browsing. Matsuo et al. (1975) proposed an algorithm that can use a single document to extract the keywords without using a corpus. They identify the high frequency terms in the document and consider their co-occurrences. A high co-occurrence means that the two keywords will always appear in the same sentence. Then they use the co-occurrence matrix to evaluate the relationship between each of the two keywords and calculate the importance score of each keyword. By using this method, they can easily extract the keywords from a single Webpage or document without undertaking a pre-training process. However, the time complexity of this algorithm is very high, especially for very long articles. One must extract each keyword and then record their positions.

Keyword density refers to the percentage of times that a keyword appears in a document or Webpage compared to the total number of words in the document. Islam et al. (2008) proposed a method that uses the random walk model to evaluate the position of terms in documents. The random walk algorithm uses a voting mechanism, and those nodes can vote for one another to determine their importance. Mihalcea and Tarau used a PageRank algorithm to extract keywords. PageRank is a link analysis algorithm used by the Google search engine, which assigns a numerical weighting to each element of a hyperlinked set of documents. Mihalcea and Tarau constructed a graph by adding a vertex for each sentence in the text, and then used the inter-connections of the sentences to represent the edges between vertices. After that, they used a PageRank algorithm to identify sentences with the highest scores as the most important sentences in a text. The keyword density methods are fully unsupervised and require no training data or pre-collection process.

2) Machine Learning Approach

Keyword extraction can use supervised learning from the examples to extract keywords. Thus, using the well-trained model, one can extract keywords from new data. Frank et al. (1999) used the Naïve Bayes learning algorithm to extract key phrases. They treated the whole extraction process as a classification task. They first generated the candidate phrases for model training. They used the TF-IDF score of the phrase and the distance into the document of the phrase’s first appearance to measure the keyphrases; then they used it to build the training model. Using this method, one can compute the probability of each key phrase and rank them all according to probability of occurrence. Turney (2000) used the C4.5 decision tree
algorithm to train and classify the positive or negative phrases. C4.5 builds decision trees from a set of training data using the concept of information entropy. At each node of the tree, C4.5 chooses one attribute of the data as the most effective separation point, and its set of samples can be split into subsets enriched in one class or the other. In this way, C4.5 uses a set of training data as input in which cases are represented as feature vectors. By this method, one can find the most representative key phrase and use it to represent the entire document.

The hidden Markov models (HMM) can also be used to extract keywords from documents. It can model a wide range of time series data and has been applied to many problem areas such as POS tagging and NP-chunking, which was mentioned in the previous paragraph. Seymore et al. (1999) wanted to label each word with a header such as title, author, or date. This system can identify the words’ class-specific unigram distribution and the probability of state transition from the training data. This system is used to extract the most important fields from the headers of computer science papers and has been proven to achieve high accuracy. Takasu (2003) enhanced the HMM algorithm into a new system, called DVHMM, which can extract bibliographic attributes from optical character recognition (OCR) processed reference strings. This system can use OCR to automatically transform the image of the documents into a text file and then extract the important keywords as an attribute. The DVHMM algorithm can estimate the probability of the recognizer’s error patterns and can be trained, using non-aligned pairs of training data.

Support vector machines (SVM) is a kind of supervised learning algorithm that is used for keyword classification in the information retrieval domain. One can use samples to train the SVM model and then divide the sample space into two distinct areas. Thus, when new data is added to the sample space, the system can easily decide to which group the data belongs. Thus, one can manually select which keywords are important and which are unimportant, and then use the SVM algorithm to create a prediction model that can be used to predict potentially important keywords to provide to the user.

Kudoh et al. (2000) used SVM to create chunk identification. They thought that using the SVM approach could achieve a high generalization performance by the feature vectors and avoid the overfitting problem. They used all POS-tagged words for training without any cutoff threshold, and the performance was still very good.

McNamee et al. (2002) proposed a named-entity tagging system that used minimal linguistic knowledge to train the SVM model. They also proved that this system could be used with new languages without too much adaptation. They built eight classifiers, and the features and leaning parameters were totally different in each of the classifiers. They explored whether separate classifiers could be combined to solve different portions of the problem. The system can detect whether or not a given token should be labeled at all, and thus the system’s overall performance will be improved.

These methods can be used only in specific domains of knowledge—for instance, computer science. If they are used in other domains, such as medical, it is impossible for the system to extract the exact keyword. Therefore, whenever one changes from one domain to another, the system must relearn and establish the model (Jones, 1999; Sebastiani, 2002).

3) Semantic Similarity
Semantic similarity is a concept that is used to measure the amount of similarity between two words or sentences within a language or a document. One can use these similarities to construct a semantic net or to measure the importance of keywords for advanced document clustering and for re-ranking search results.

A. Document-based model
The latent semantic analysis (LSA) proposed by Deerwester et al. (1990) can measure the relationships of word-word and word-passage, and it has been discussed in many natural language processing research articles. This process can analyze documents to find the underlying concepts within the documents. In this way, both a document and the query entered by the user all become column vectors. Then these two vectors can be compared to find whether or not each word means only one concept and each concept can be describe by only one word. This method basically considers only the co-occurrence of keywords as vectors and then compares them. It does not measure the sequence or affiliation between the words. Hofmann et al. (1999) adapted the LSA algorithm and proposed the probabilistic latent semantic analysis (PLSA). This method can find the relationship of topics that are associated with terms
and documents and can solve problems related to polysemy and synonymy. This algorithm can be used to group those documents that have the same topic. The PLSA’s flaw is the same as the LSA’s: it does not consider the order of the sequence. Matveeva et al. (2005) proposed generalized latent semantic analysis (GLSA), which uses a large corpus to compute the term vectors for the vocabulary of a document collection. This algorithm creates a word-by-word matrix so that if there are a great many potential keywords, the computational cost will increase.

Hyperspace analogue to language (HAL) views context as only the words that immediately surround a given word. It measures the co-occurrence of the words in the repository and creates a word-by-word matrix to represent the term vector. Song and Bruza (2001) used HAL to generate higher order concepts and to determine the concept inclusion and compute the concept composition. Azzopardi et al. (2005) used the probabilistic HAL method, which uses probabilities to represent the co-occurrence of words. Its performance is better than the traditional HAL algorithm. Lindsey et al. (2008) proposed the best path length on a semantic self-organizing map (BLOSSOM) method, which calculates the semantic distance from one keyword to another, like the traversal algorithm, and uses the SOM algorithm to reduce the dimensions. Each pair of keywords will have a similarity score so that a node selection method can be used to find the shortest path on the undirected graph to form a concept-path. They proved that this algorithm can achieve high performance when using ontology-based dimension words.

B. Vocabulary-based model

WordNet groups words into synsets (synonym sets) and records the semantic relations of these synsets. This method was first developed by George A. Miller and others in 1985. Currently, WordNet contains 117,659 synsets and 206,941 word-sense pairs. Banerjee and Pedersen (2002) presented a Lesk’s dictionary-based word sense disambiguation algorithm, which used the WordNet as a lexical database. The Lesk algorithm is based on the assumption that words in a given neighborhood will tend to share a common topic. Pederssen et al. (2004) used the WordNet to measure the semantic similarity between each pair of synsets. Thus, if two concepts are inputted to the system, it will provide their similarity degree.

Explicit semantic analysis (ESA), which has been proposed by Gabrilovich and Markovitch (2007), is a measure that uses Wikipedia as a dataset to compute the semantic relatedness between two arbitrary keywords. Because of this, Wikipedia has become the largest knowledge repository on the Web. They used the TF-IDF algorithm to weight the extracted keywords to form the vectors of the concept. By using ESA, the correlation of computed relatedness scores with humans and texts have all been greatly improved.

C. Worldwide Web-based model

The basic idea of this kind of algorithm is to use the Worldwide Web as the largest corpus in the world. This model can overcome the restrictions of a limited corpus so that information can be extracted from different knowledge domains. Normalized Google Distance (NGD) was proposed by Cilibrasi et al. (2007), and it has been used to calculate the relationship between two words. By using Google as the corpus, one does not have to pre-collect the data and construct the corpus from scratch. Instead, each keyword can be entered into the search engine, and the number of search results can be used to calculate the similarity of the the keywords. Makrehchi (2007) proposed an automatic taxonomy extraction algorithm, which used the NGD algorithm to measure the dependency of two terms and then transform the term dependency matrix into an adjacency matrix, which can form the taxonomy matrix.

The NGD algorithm can also be used in document clustering. Batra and Bawa (2009) used the NSS to evaluate words in pre-defined categories and to discover the Web services semantically. Normalized similarity score (NSS) is derived from the NGD algorithm. Some of the pre-defined categories include zip code, temperature, weather, and the like, and the system will extract terms to calculate the most relevant documents and allocate the service to it.

3. Applications

1) Word Sequences

As we know, the Google search engine can recognize the sequential order of keywords. Thus, if several keywords are entered into the search engine in different sequences, the search results for each will greatly differ. Also, if one wants to compare
the similarities between Webpages and documents, algorithms can be used to extract the keyword sequence and similarity measure methods can be used to cluster related information, as shown in figure 3.

\[ A1 \rightarrow B1 \rightarrow C1 \rightarrow D1 \]
\[ A2 \rightarrow B2 \rightarrow C2 \]
\[ A3 \rightarrow B3 \]

**Fig. 3:** Keyword sequence and document clustering.

### A. N-gram

An n-gram is a set of n consecutive characters extracted from a word. This method was proposed by Freund and Willett (1982), and it relies on the likelihood of sequences of words, such as word pairs (in the case of bigrams) or word triples (in the case of trigrams). Papineni et al. (2001) proposed an n-gram co-occurrence scoring procedure, named BLEU, to measure the accuracy of machine translations. It calculates the similarities between the reference text and candidate translations. The output of the BLEU is always a number between 0 and 1. This value indicates the degree of similarity between the candidate and the reference texts. Lin and Hovy (2003) evaluated the BLEU and found that unigram co-occurrence statistics is a good automatic scoring metric. Also, more extensive n-grams tend to score for grammaticality rather than for content.

### B. Multiple keyword sequence extraction

Sato and Saito (2002) proposed a method of extracting word sequences by using a support vector machine (SVM). They used this method to find relationships in documents. Its original design was intended to obtain the least-biased statistical inference when there was insufficient information, but it can also be used to extract bilingual word sequences. Feng and Croft (2001) used the maximum entropy (ME) algorithm to extract English noun phrases automatically. They used probability methods to find a sequence of keywords from one word to the next word and also used it to realize article summarization. Li et al. (2003) used the same idea to extract a sequence of keywords from the news and clustered those related pieces of information to provide the user with a summarization of search results. But they also said that the ME model required a large training data to evaluate the feature parameters.

#### 2) Keyword Expansion

Jansen et al. (1998) found that most users enter only approximately 2.35 terms in the search engine. Also, most users have not been satisfied with the
search results provided by search engines. One reason is that many users lack domain knowledge on how to enter a precise keyword to describe their thoughts, not to mention entering a meaningful keyword sequence in the search engine. Another problem is that the order in which keywords are entered will also affect the results.

Fortunately, the Google search engine provides a keyword expansion suggestion to help the user enter more keywords in order to indicate what they are actually trying to find. When a keyword is entered into the search box, the search engine provides relevant next-keywords, as depicted in figure 4.

![Fig. 4: Google keyword expansion.](image)

This method can be useful if the user has sufficient knowledge of the keyword sets that Google provides. Basically, it represents a kind of collaborative recommendation so the information provided is not customized and may not satisfy the users’ needs. Bharat (2001) proposed a system called “SearchPad,” which works collaboratively with results pages and can simultaneously search AltaVista, Excite, Google and Hotbot. This system allows users to remember queries and associated leads by the use of a convenient helper window. Thus, the system can discern which results will be most valuable to users.

Finkelstein et al. (2001) proposed a system that extracts the context around the text that the user highlights. Then the information is sent to a server site for analysis to find the most important context words in order to advance the selected words to a more meaningful keyword set. In the end, the results are dispatched to several search engines in order to provide the user with more relevant information.

Liu et al. (2004) proposed a personalized Web search system that makes use of user profiles and general profiles to improve retrieval effectiveness. General profiles here means mapping user queries to a set of categories. The authors constructed it based on the open directory project (ODP) category hierarchy. They combined these two profiles to form the user’s search intention. When a user inputs a keyword to the search engine, it will return several categories, allowing the user to choose the category that is most likely to contain the information sought and thus obtain more accurate information.

3) User Profiles
Most personalization methods focus on creating a single huge profile that can respond to all users’
queries. Others focus on creating a personal user profile that represents a single user’s interests. In this manner information can be collected from user clicks and then used to infer users’ interests. This system also uses a personalized agglomerative clustering system, which creates more meticulous personal query clusters and allows the search engine to provide the user with customized search results.

Cui et al. (2003) proposed a log-based query expansion that assumes a user will select the relevant information about the document being read. Thus, the researchers could calculate similarities between the users’ interests and the keywords entered. For example, if the system can detect and conclude that a user is a basketball fan, when the keyword “Michael” is entered in the search engine, then it will provide search results for Michael Jordan instead of Michael Jackson.

Zhang and Nasraoui (2006) analyzed and modeled the users’ sequential search behavior. They combined this method with a traditional content-based similarity method, which despite its simplicity makes it effective in finding related queries. The user profile can also analyze the local context, based on the top-ranked documents initially retrieved for a given query, and then add the best-scoring concepts to the query (Xu and Croft, 2000). In this way, keywords can be expanded by using the co-occurrences of the global and local terms, thereby obtaining more accurate search results. However, this method does not consider semantic relationships between keywords, and the top-ranked documents for each search take a great deal of time to collect.

Some researchers have tried to add the semantic relationship of the keywords, using ontology as a solution. Thus, in order to find information about Michael Jackson, a user must enter both “Michael Jackson” and “singer” in the system. This allows the system to draw from its ontology and use the concept-level keyword to search and obtain more accurate search results (Khan et al. 2004).

A major weakness of using the user profile is that enormous amounts of data must be collected about each user, and the constructed models are quite large, thus making the computational time extremely long. Also, if a keyword that was not previously connected to this knowledge domain is used, then the user profile method will be unable to provide any useful information.

---

**Diagram:**

- **User:**
  - Reading
  - Browsing

- **Community:**
  - Reading
  - Browsing

- **Interpersonal interaction on the Internet:**
- **Model comparison:**
- **Multiple user profiles**
4) Collaborative Filtering (CF) Recommendation
Adomavicius and Tuzhilin (2005) have presented various recommendation systems and classified them into three main groups: content-based, collaborative, and hybrid. Figure 5 is a graphic illustration of the recommendation system. The content-based method basically uses a single user profile to recommend the related information to the user. The collaborative filtering recommendation method requires a large amount of information and a significant computational effort; hence, only large companies or groups use this technique to help users expand their keywords. Its basic function is to collect all the users’ search interests or input keywords and try to find other users who have the same interests. Therefore, if a keyword is entered in the search engine, the system uses related searching information about the community member to provide a query recommendation.

The hybrid method is a content-based method, and it cooperates with collaborative methods. In this way, the recommendation results for the hybrid method are more accurate than for those using only pure methods to retrieve information. Konstas et al. (2009) used social networks to implement a collaborative recommendation. They showed that a social network can provide additional knowledge that improves the performance of the system. They suggest that companies can obtain greater profit by encouraging users to create social networks.

There are three main grainuairties of CF similarity measurements: user-based, item-based, and association-rule. User-based CF first estimates users’ reading behavior and uses the vectors to represent users’ interests. Thus, the similarity between those vectors and groups of users with similar interests can be calculated (Resnick et al., 1994; Breese et al., 1998). Item-based CF focuses on item similarity rather than user similarity. It records the items in which users are interested and then selects the k-most similar items as the vectors to calculate similarity (Sarwar et al., 2001). Association-rule CF uses the data mining techniques to find the potential relationships between the items (Smyth et al., 2002). The difference between item-based measurement and association-rule is that item-based methods do not have a rule generation process, but instead focus only on the relations between item and item.

Some researchers think that the CF method can be easily attacked by an injection attacker that uses a lot of unrelated documents to bias the recommendation results. O’Mahony et al. (2004) analyzed the robustness of collaborative recommendation and presented a framework to examine the stability of recommendations. They believed that accuracy and stability are the two main aspects of robustness. Sandvig et al. (2007) tried to explore the robustness of a recommendation algorithm that is based on an association rule to capture the co-occurrence pattern in user profiles. Their system also produced significant improvement in stability.

We believe that people in the same community will likely have similar interests, so when they attempt to search for something outside their knowledge domain, the search results provided by CF algorithms will be disappointing. Also the computational complexity and data repository is a major problem for using this method.

5) Re-ranking the Search Results
Google uses a page-rank algorithm to weight each Webpage and uses a keyword-based algorithm to rank the search results. This helps the user find the most important information about the inputted keyword efficiently. If the user lacks adequate domain knowledge about how to enter precise keywords in the search engine, the results will be generally disappointing. Most researchers use user profiles to build browsing behavior models that reflect the users’ interests. These models can be used to re-rank the search results in order to deliver the most relevant results to the user. To refine the search results, Zhuang and Cucerzan (2006) proposed a method that uses query logs to construct a query context. Leung and Lee (2008) proposed a system that extracts concepts from the search results and uses the relationships between concepts and queries to enhance the accuracy of the search engine.

Another basic problem concerns how to measure the ranking of the search results. Bar-Ilan et al. (2006) used various measures to compare the search results of different search engines. Spearman’s foottule is the most famous measurement tool and has been discussed in many research articles. We can compare the re-ranking results provided by the system with the results of
re-ranking by the experts to see whether or not the system can provide more relevant information to the users.

4. New Applications for Information Retrieval

1) Keyword Suggestions To Improve User Browsing Behavior

In Wikipedia, the editor of the Webpage can identify the important keywords and create a link for those keywords to their related Webpages. In this way, the user can easily find and read an article and obtain more useful advanced information from the reading. Mihalcea and Csomai (2007) proposed a system that uses Wikipedia as a resource for automatic keyword extraction and provides the user with a link to those keywords. In this system, when a document is input to it, the system can identify the important concepts and automatically link these concepts to the relevant Webpages in Wikipedia. The system measures the importance of phrases by using the TF-IDF algorithm and then uses a Chi-square independence test to find co-occurrences of two events. This method allows one to rank those keywords or phrases and automatically provide useful information to the user without the benefit of a pre-constructed semantic corpus. However, the TF-IDF methods require pre-collected documents or Webpage sets to calculate the inverse document frequency (IDF) values. In Web information retrieval, all executions should be conducted online and in real-time because otherwise it requires too much time to download these data to find the IDF value.

In our previous work (Chen and Lin, 2010), we wanted to find a way to provide immediate information to users when they are reading an article or browsing a Webpage. Our original idea was to construct a new system that could improve the reading behavior of users. The system would automatically extract the keywords from the article and use an NGD algorithm to measure the relationship between any two keywords. Thus, all articles could be transformed into a Wiki-like system and there would be no need to enter each keyword in Google to find information about a keyword. The system architecture is illustrated in figure 6. We used the NGD algorithm to construct the system to eliminate the need to pre-collect information from the Web. All we needed was simply to obtain the number of search results for each keyword, which then produced important information to improve the efficiency of users’ reading. In this manner, we were able to easily identify important keywords from articles the users were reading to provide them useful information about it.

2) Document Clustering and Web Taxonomy

Many articles focus on this research domain, and the methods used are all based on those we discussed in the previous section to extract keyword sequences. The following three methods are used to cluster similar documents in the same group: vector space model, frequent itemset, and K-means and agglomerative. Also, by using the document
clustering algorithm, we can determine the users’ reading or browsing behaviors and then use this information to customize the search results that are most relevant to the user.

A. Vector space model

Salton et al. (1975) proposed the vector space model for automatic indexing. They thought that the document space will comprise one or more index terms, and those terms can be weighted according to level of importance. If one obtains the index vectors for two documents, the similarity between them can be easily calculated, and those documents with the highest similarity scores can be clustered. Most researchers use a cosine measurement to compute the cosine of the angle between these two vectors.

Jaccard similarity coefficient (Jaccard index) is a statistic used to compare the similarity and diversity of sample sets. It compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the share terms. Luo et al. (2009) used the cosine and Jaccard similarity coefficient to measure similarity to coordinate with the K-means algorithm for document clustering. They found that the cosine performs better than the Jaccard index for document clustering.

Pearson’s correlation coefficient can be used to measure the correlation between two objects on all attributes. Shardanand and Maes (1995) used the standard Pearson r correlation coefficient to measure similarity between user profiles. Yang et al. (2002) thought that strong coherence may exist only on a subset of dimensions so that sometimes the correlation value will not be very high. The accuracy of this algorithm will be affected by the data because it lacks the ability to deal with attribute bias.

B. Frequent itemset mining

A frequent itemset is a set of frequently occurring items whose probability of co-occurrence in the database will be higher than that of the threshold. Edith et al. (2006) proposed a method that used the maximum frequent itemset sequence that was not a subsequence of any frequent sequence as the representative term vector of documents. They then used the k-means algorithm to cluster similar documents and group them together.

Fung et al. (2003) used frequent words and the hierarchical clustering algorithm to cluster related information. Thus, users can easily obtain information by using the meaningful cluster labels. They thought that frequent itemset could describe something common to all documents in a cluster, and these itemsets could be used to construct clusters into a topic hierarchy to achieve high accuracy and meaningful results. Most of the previous research used the TF-IDF algorithm from which frequent words can be easily extracted. Therefore, each document can be represented by a vector of weighted frequencies. But the authors used global frequent items instead of the TF-IDF to construct the initial clusters. Finally, they used their score function to measure the goodness of the initial clusters and tried to disjoint them to obtain the cluster tree.

Li et al. (2008) uses frequent word sequence and k-mismatch for document clustering. The problem with using k-mismatch is to find all occurrences of patterns in a text with the most k mismatches and then use them to compare the similarity between each document’s frequent itemsets. They believed that the sequence of words provides more valid information than individual words and also represents the topics very well. Thus, using this method, greater accuracy can be achieved than that produced by traditional text clustering methods that use the vector space model.

C. K-means and agglomerative

K-means is the simplest method of cluster analysis that aims to partition the data into several clusters. The way it works is to define k centroids, one for each cluster, and then take each point belonging to a given dataset and associate it to the nearest centroid. The weakness of k-means is that it is not easy to determine the number of clusters, and this deeply affects the results of the classification. Another kind of clustering method is the hierarchical clustering algorithm. Most researchers use the agglomerative algorithm to cluster similar information. Agglomerative hierarchical clustering is a bottom-up clustering method that treats each document as a single cluster and then continues to calculate and merge pairs of clusters until all clusters have been merged into a single cluster that contains all related documents.

Steinbach et al. (2000) used these two algorithms to conduct a document clustering system and evaluate its accuracy. They used k-means, bisecting k-means, and hierarchical clustering and found that bisecting k-means achieve the highest accuracy.
They found that any two documents may contain many of the same words. Thus, understanding how to distinguish between these documents and allocate them into various different classes, especially the nearest-neighbor documents, is a difficult job. The agglomerative method will often put documents of the same class into the same cluster. But the k-means algorithm uses a global property approach, which computes the average similarity of all the documents to clustering so that it can overcome the mixed nearest-neighbor problem. Otherwise, the bisecting k-means also has the same advantage of the k-means, and its performance and execution results are better than those of k-means.

Web taxonomy is becoming more and more important in our daily life because it allows us to gather all kinds of information from the Internet. The most important step in constructing a taxonomy is to extract the domain-specific terminologies and determine the relationship of each of them. Webpages always contain many new terms, so a traditional knowledge database that has been created manually is not suitable since the cost of collecting the data manually is excessive. Thus, a system is needed that can perform this function automatically. Zhang and Lee (2004) used the TSVM algorithm to deal with this problem. The transductive SVM (TSVM) algorithm, first introduced by Joachims, exploits its prior knowledge to speed up performance of the classification, especially for small training examples.

Godoy and Amandi (2006) produced a document clustering algorithm that used the unsupervised concept learning over Web documents to acquire user profiles. This algorithm can extract semantic relations from a Webpage so that they can be integrated into ontology and thus provide more detailed information when searching with a search engine.

We think that the NGD-based algorithm can also be used to cluster related documents. We have successfully used the NGD algorithm combined with the PLSA to find the meaningful keyword sequences in a document. But the problem is that the relationships between single knowledge domains will be less complex than those found in multi-domains. If one attempts to cluster documents that contain different kinds of knowledge, the results will not be as good. Possibly the NGD algorithm could be adapted to create one that is more accurate in finding absolute relationships between two keywords.

3) Joint Inference
The key problem in citation matching is how to extract bibliographic information from the citation lists in technical papers and ultimately merge the relevant information from those same articles. CiteSeer is a digital library and search engine for scientific papers. It provides automated citation indexing and citation linking using the autonomous citation indexing (ACI) method. It groups citations using agglomerative clustering based on a text similarity metric. The formats of the citation are different and the titles of some articles are very similar. This makes understanding how to identify and cluster together the same citations accurately very difficult. The interesting point of this problem is finding out how to determine when two objects are describing the same thing. This technique can be used not only in academic journal databases but also to identify similar objects that are written in different formats, such as in patent mining.

In traditional information extraction, every single keyword is treated as an individual segment. Thus, certain field boundaries, like punctuation symbols, can be detected to classify each item within the correct field of the database. Poon and Domingos (2007) used the Joint Segmentation (Jnt-Seg) model to improve the accuracy of citation matching. They used the hidden Markov model (HMM) to detect field boundaries and incorporated it with the Markov logic network (MLN). In this way, the title can be clearly extracted from one citation by using the boundary of punctuation. High probability can also be used to extract similar citations that lack the clear boundaries of punctuation. The most difficult problem of joint inference is that many rules must be predefined in order to use MLN to improve the accuracy of the system. Research in this area is limited, but it may be possible to design a new algorithm that can also use Google as the corpus to combine with the association rule instead of setting new rules individually.

4) Knowledge Map and Sentimental Analysis
Wesley Vestal (2003) wrote this: “Knowledge mapping is a process by which organisations can identify and categorise knowledge assets within their organisation—people, processes, content, and technology.” Browne et al. (1997) conducted an experiment that showed that knowledge mapping can elicit high-quality information for decision making because it allows graphic information to be
immediately connected to propositional information to increase the possibility of a fully analytic-minded result.

Lin et al. (2006) proposed a knowledge mapping management system to facilitate knowledge management in virtual communities of practice. A virtual community is a social network of individuals who interact through a specific medium, especially the Internet, to pursue mutual interests or goals. The virtual community contains many different knowledge domains of information. The system that Lin proposed can automatically categorize knowledge from documents in these virtual communities. Then, it can create a knowledge map and use up-to-date information to maintain the knowledge map. But the text indexing process must pre-collect the documents and then use them to form a knowledge map.

Our latest research is about how to use the NGD algorithm to construct a knowledge map. Our previous work has shown that these kinds of algorithms can be used to extract the most important keywords and also can obtain a relative score for two keywords. Thus, we can use it to construct a knowledge map automatically and quickly. If a user is reading an article or a Webpage, the knowledge map can be provided to the user within a few minutes for decision making or advanced reading. The knowledge map will use hierarchical architecture, which is also generated on-line in real-time without any training data or pre-collecting process.

By using the knowledge map, one can understand not only the main information about an article, but can also extend that information to sentiment analysis. The aim of sentiment analysis is to determine the attitude of a writer with respect to a given topic. Pang and Lee (2004) proposed a method that applied the machine-learning algorithm to categorization techniques. They used the SVM and Naïve Bayes algorithm to train and construct the subjective detectors. They demonstrated that this method can successfully preserve the sentiment information in the document. We propose to use the knowledge map that was mentioned before to identify the most representative subgraph of the knowledge map as the subjective object. We will then attempt to find a new way to calculate the degree of user preference. By doing this, we will be able to immediately understand the author’s preference. This should prove very useful if we want to know a brand image or individual strength, because we can use the agent to find all the Webpages on which this brand is mentioned, and then this system will calculate and find the opinion on the Internet.

5. Conclusions

In this paper, we have summarized some of the most important techniques in the IR domain and introduced an algorithm that can measure the relationships between two keywords or keyword sequences. We believe that the traditional IR algorithm will be useless in the near future because of the information provider is from the personal documents or the company’s database to the Internet. Our recent work has proved that by using NGD-like algorithms, we can extract keywords and determine their importance instead of using the traditional TF-IDF algorithm to do this. We also combined it with the PLSA so that we could extract a meaningful sequence of words to represent the meaning of a paragraph. We are trying to use these sequences to implement document clustering and knowledge map construction. All executions are conducted in real-time without any data collection or training procedures. Our hope is that one day when we read an article or a Webpage, the system will be able to provide information on important keywords to facilitate our reading. It will also be able to determine whether any related articles on the topic are available and provide a summary of them.

Another proposal is to invent a system that can provide immediate suggestions on written works to an author. For example, if one is writing about related works from a research article and keys in the phrase, “The latent semantic analysis (LSA) which is proposed by Deerwester et al. (1990),” the system will be able to detect two keywords, “latent semantic analysis” and “Deerwester.” In this way, the system can use the algorithms to find articles about these two keywords and provide useful information for writing other articles. We also plan to use the same methods to conduct sentiment analysis. We can immediately discern the preferences of the Blog or Webpage writer, and that will help us analyze market reactions to product sales or government policies.

ACKNOWLEDGEMENTS

This study is conducted under the “III Innovative
and Prospective Technologies Industry Project” of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of the Republic of China.

References


Proceedings of AAAI’99 workshop on machine learning for information extraction.


