

Neural Based Approach to Keyword Extraction from Documents

Taeho Jo

SITE, University of Ottawa, 800 King Edward Ave
Ottawa, Ontario Canada K1N 6N5
tjo018@site.uottawa.ca

Abstract. Documents are unstructured data consisting of natural language. Document surrogate means the structured data converted from original documents to process them in computer systems. Document surrogate is usually represented into a list of words. Because not all words in a document reflect its content, it is necessary to select important words related with its content among them. Such important words are called keywords and they are selected with a particular equation based on TF (Term Frequency) and IDF (inverted Document Frequency). Actually, not only TF and IDF but also the position of each word in the document and the inclusion of the word in the title should be considered to select keywords among words contained in the text. The equation based on these factors gets too complicate to be applied to the selection of keywords. This paper proposes the neural network model, back propagation, in which these factors are used as the features and feature vectors are generated, and with which keywords are selected. This paper will show that back-propagation outperforms the equation in distinguishing keywords.

1 Introduction

Information systems dealing with documents, such as KMS (Knowledge Management System), IRS (Information Retrieval System), and DLS (Digital Library System), requires the storage of not only documents also structured data, called document surrogate, associated with documents. Documents are unstructured data, which are written in natural languages and are not able to be processed directly by computer. A typical kind of document surrogates, which are the converted structured data process directly by computer, is indices of documents including main words reflecting their contents.

Indexing defines the process of converting a document into a list of words included in it. The result of indexing, a list of words is called index language [1]. Not all words are related with the content of a document. The words, which play the grammatical roles in the document and are not related with its content, are called stop words. These words correspond to preposition, conjunction, special verbs (be, can, may, should, and so on), pronoun, and particles in English. In general, words corresponding to noun and verb reflect the content of the document. Therefore, stop words should be excluded in the process of indexing.

Not all words except stop words in the document are related with the content of the document, either. In other words, all words except stop words need not to be included in the index language to process documents more efficiently. For example, in the case of news articles covering broad areas including politics, sports, and information technology, the word, “computer”, may be very important word to categorize them. But in the case of documents dealing with specific areas within information technology, the word, “computer”, is very trivial word discriminating them within computer. The technologies of determining whether a word should be included in index language are required to process documents.

The words actually related with the content of the document are called informative words or keywords. These words should be included in the index language associated with a document, and the others should be excluded. The discrimination of words except stop words reinforces the function of search engine in information retrieval system; it prevents from retrieving non-necessary documents. This step in indexing documents provides benefits functions of text mining. In the process of text categorization, this step prevents the misclassification of each document caused from nouns or verbs not related with the content of the document. In the process of text summarization, it is expected to prevent from selecting sentences weakly related with the content of the document.

The researches about the identification of keywords in the given document have been one of important research issues in information retrieval for long time. In 1988, G. Slaton proposed weighting words included in documents in first time [2]. In 1993, F. Pereira proposed to cluster words based on their frequency in the document [3]. In 1995, Y. Yang insisted that the identification of keywords is necessary for text categorization as features [4]. E. D. Wiener enumerated the schemes of selecting keywords in his Master thesis [5]. In 1997, M.E. Maron proposed a scheme of the identification of keywords based on the relevance of document [6]. In 1998, Y. Tseng applied the schemes of selecting multilingual keywords [7]. In 1999, T. Hofmann proposed a scheme of extracting keywords, called PLSI (Probability Latent Semantic Indexing), which is a scheme improved from LSI (Latent Semantic Indexing). M. R. Brent applied unsupervised learning algorithms to cluster words included in the given document [8]. S. Soderland developed the system called WHISK, in which not only keywords also important information are extracted from documents by rules [9]. In 2000, D. Freitag proposed the combination of schemes of generating keywords and important information from the given document [10]. These researches show that the identification of keywords is very important in both information retrieval and text mining. The identification of keywords has been developed based on TF (Term Frequency) and IDF (Inverted Document Frequency) in these researches

This paper includes three sections after this section: the application of back propagation to keyword selection, experiment and results, and discussion. Next section will describe how a most common model of neural networks, back propagation is applied to the selection of keywords; how features of words and output are defined. In third section, back propagation will present the better result than two equation models used in information retrieval in the precision of selecting keywords. The last section will discuss the limit in applying the back propagation to keyword selection and the remaining tasks to validate the comparison between back propagation and mathematics equation based on TF and IDF more clearly.

2 The Application of Back Propagation to Extract Keywords

This section will describe the design of a neural network model, back propagation, for judging whether a word is keyword or not. In the field of pattern classification, it is assumed that an entity should be represented into a numerical vector. Before applying neural-based or statistical approaches, it is important to decide features for the classification. The algorithm of training and generalizing back propagation mentioned in this paper is explained in [11]; this paper will skip its algorithm. Features for judging whether a word is keyword, will be defined, and the architecture of the back propagation designed in judging keywords, will be presented in this section

Before determining feature values, a group of documents are given as a sample. These documents are called sample documents [12], which are heterogeneous documents in their contents. Collecting various documents at random makes this group of sample documents. Sample documents are necessary to determine main two features of each word: IDF (Inverted Document Frequency) and ITF (Inverted Term Frequency).

The input features of each word in the given document are like the following this. The features, IDF and ITF, requires sample documents enough to maintain the robustness of the system, before computing their value of each word.. The features, TF, IDF, and ITF, are represented in integers greater than or equal to zero, while the others, T, FS, and LS, are represented in binary values: zero or one; for example, if the word is in the title of the document, T is one, otherwise T is zero.

- TF (Term Frequency): The frequency of the word in the given document
- IDF (Inverted Document Frequency): The number of documents including the word in sample documents prepared before
- ITF (Inverted Term Frequency): Total frequency of the word in sample documents before
- T (Title): Existence of the word in the title of the given document
- FS (First Sentence): Existence of the word in the first sentence in the given document
- LS (Last Sentence): Existence of the word in the last sentence in the given document

Output features each word are given like the following this. Both features are represented in binary values

- K (Keyword): If the word is judged to keyword, K is one, otherwise, zero
- N (Non-keyword): If the word is judge to non-keyword, K is one, otherwise zero.

Therefore, the architecture of back propagation is designed like the figure 1. The input features are TF, IDF, ITF, T, FS, and LS as mentioned above; the number of input nodes is six. The output features are K and NK as mentioned above; the number of output nodes is two. The number of hidden nodes is determined arbitrary; there is no absolute rule to set the number of hidden nodes.

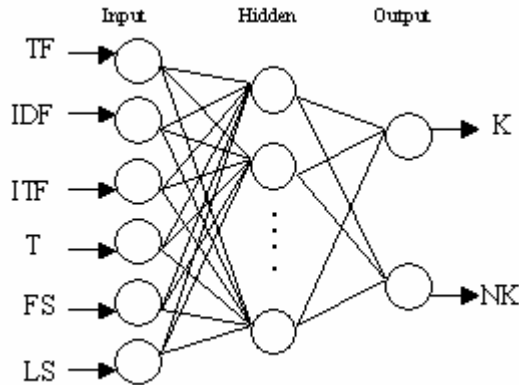


Fig. 1. The Architecture of Back Propagation to Judge Keywords

3 Experiment & Results

In this section, the result of the application of back propagation to judge keywords will be presented. The data of test bed is the collection of news articles. The neural based approach to judge keywords is compared with equations based on TF (Term Frequency) and IDF (Inverse Document Frequency) in the performance of precision. The sample collection of news articles is required to set the value of IDF (Inverse Document Frequency) of each word contained in a particular news article. The domains of news articles are various and news articles were selected from the site, www.newspage.com in 1999. The total number of sample news article is none hundreds to determine the value of the IDF of each word. The domains of sample news articles are like the following this. The number of news articles in sample selection to determine the value of IDF of each word is one hundred per domain and the number of domains is nine.

- Business
- Healthcare Regulation
- Migration
- Pharmacy
- Politics
- Public
- Sports
- Wireless communication
- WeirdNuz

In this experiment, the proposed approach is compared with two equations assigning the weights of terms included in the document. Both equations for computing weights of words are functions of TF (Term Frequency) and IDF (Inverse Document Frequency). The equation (1) is the most common equation to compute the weight of each term in information retrieval [12]. The equation (2) is the equation used in the project of developing two modules, text summarization and text categorization, in the

knowledge management system called KWave as the name of the product in the vendor, Samsung SDS [18]. In equation (1), N is the number of sample documents and its value in this experiment is set to 900, because the number of sample documents is 900 as mentioned above. ITF (Inverse Term Frequency) in the equation (2) is the total frequency of the word in the sample documents, while TF is the frequency of the word in the given document.

$$W_i = TF(\log_2 N - \log_2 IDF + 1) \text{----(1)}$$

$$W_i = \frac{TF^m}{(IDF + ITF + 1)^n} \text{----(2)}$$

The precision of selecting keyword with two above equations is illustrated in the table 1. The selection rate in the table 1 is the rate of words selected as keywords of the document to total words included in it. In the table 1, the result with equation (1) is better than that with equation (2). This fact is why the equation (1) is most popular in the field of information retrieval.

Table 1. Precision with Equations of TF and IDF

Selection Rate	Equation (1)	Equation (2)
20%	0.7	0.68
40%	0.82	0.76
50%	0.8	0.74
60%	0.78	0.72
80%	0.7	0.68

The precision of judging keywords in documents with back propagation, a model of neural network, is illustrated with the table 2. As mentioned in the previous section, the learning algorithm of a model of neural network, back propagation, is detailed in the textbook about neural network such as the literature [11]. Each column means the number of epochs in training the back propagation, and the row means the threshold for the binary classification, whether it is a keyword or not. In the table 2, if 0.51 and 0.49 are given as the thresholds for the binary classification and the back propagation is trained with 7500 epochs, the precision is maximum, 0.92. Like the table 2, the back propagation is suitable for the classification based multiple factors; the back propagation presents better performance in the judgment of keywords with the optimized parameters.

Table 2. Precision with Back Propagation

	2500	5000	7500	10000
0.9	0.58	0.58	0.58	0.62
0.8	0.7	0.68	0.74	0.68
0.7	0.72	0.7	0.74	0.76
0.51	0.8	0.84	0.92	0.84

4 Conclusion

This paper proposed the application of back propagation and consideration of more factors with the addition to TF (Term Frequency) and IDF (Inverse Document Frequency). In the previous section, back propagation presented better result than two equations based on TF and IDF: the equation (1) and the equation (2). But if the thresholds are set strictly, such as 0.9 and 0.1, the precision is worse than that of two equations. The reason is that the actual output of the word is between two thresholds and such word is classified neither a keyword nor a non-keyword. Two equations don't contain non-classified words, but misclassified words. In two equations, the word is correctly classified or wrong classified, while in the back propagation, the word is correctly classified, non-classified, or misclassified. If two thresholds are set to 0.51 and 0.49, the precision with the back propagation is better than that with two equations.

In this paper, the application of the back propagation to the judgment of keywords is validated restrictedly. The definition of the back propagation to the judgment of keywords may be considered in various ways. The words in a particular document need to be refined instead of keyword or non-keyword. The proposed approach of the judgment of keywords should be compared with more complicated approaches. It is to discover discriminators to judge whether a word is a keyword of the document or not with the statistical analysis of postings of words.

References

1. Korfhage, R. R., Information Storage and Retrieval, John Wiley & Sons Inc (1997)
2. Salton, G. and Buckley, C.: Term Weighting Approaches in Automatic Text Retrieval. *Information Processing & Management*. 24 (1988) 513-523
3. Pereira F., Tishby, N., and Lee, L.: Distributional Clustering of English Words. The Proceedings of 30th Annual Meeting of the Association for Computational Linguistics, (1993) 183-190
4. Yang, Y.: Noise Reduction in a Statistical Approaches to Text Categorization. The Proceedings of SIGIR 95, (1995) 256-263
5. Wiener, E. D.: A Neural Network Approach to Topic Spotting in Text. Thesis of the Graduate School of the University of Colorado, (1995)
6. Maron, M. E.: Probabilistic Indexing and Information Retrieval. In: Sparck, K. and Willett, P. (eds.): *Readings in Information Retrieval*. Readings in Information Retrieval (1997) 39-46
7. Tseng, Y.: Multilingual Keyword Extraction for Term Suggestion. The Proceedings of SIGIR 98, (1998) 377-378
8. Hofmann, T.: Probabilistic latent indexing. The Proceedings of SIGIR 99, (1999) 50-57.
9. Soderland, S.: Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning* 34. (1999) 233-272
10. Freitag, D.: Machine Learning for Information Extraction in Informal Domains. *Machine Learning* 39 (2000) 169-202
11. Freeman, J. A. and Skapura, D.M.: *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison-Wesley Publishing Company (1992)
12. Korfhage, R.R.: Information Storage and Retrieval. John Wiley & Sons Inc (1997)
13. Jo, T.: The Application of Text Mining to Knowledge Management System, Kwave. white paper in Samsung SDS, (1998).