Study on Root + Affix Form-Based Mongolian Information Retrieval Unit

Junying Yue[1], Guanglai Gao[2], Min Lin[1]

[1]Inner Mongolia Normal University, School of Computer and Information Engineering, Hohhot 010022 ;

[2] Inner Mongolia University, School of Computer Science, Hohhot 010020

hunter2011@foxmail.com

Keywords: Mongolian information retrieval; Root + affix form; Retrieval unit; Structured query; lemur language model

Abstract. In order to improve the efficiency of Mongolian information retrieval, further research is carried out on root + affixes form-based retrieval unit with selected information retrieval model by combining the characteristics of Mongolian language. Selectable information retrieval model include TF-IDF Model, Vector Space Model and Lemur Language Model. The following four steps are conducted for root + affixes form: establishment of index for corpus, query analysis, retrieval and evaluation. Thereby, comparison is conducted on recall rate and precision rate to find out the proper retrieval unit. The results show that root + 2 affixes is the proper retrieval unit for Mongolian information retrieval system.

Introduction

Information retrieval technology was originated from 1950s. Currently, English and Chinese information retrieval technologies are relatively mature. With the Mongolian information develops in the form of website, it provides the applicable foundation for Mongolian information retrieval technology.

Study on Mongolian information retrieval model to improve retrieval performance is the commonly used method by researchers. Hierarchical Mongolian language model [1] divides Mongolian language model into three levels according to the characteristics of typical configuration affixes of Mongolian language, i.e., stem and reliance on stem; stem and affix reliance; affix and affix reliance. Based on the three levels, proper language models are constructed respectively and combined into a complete language model suitable for Mongolian language. Analysis is conducted for trigger pair characteristics based on the long-distance Mongolian language model of trigger pair so as to design the model for long-distance Mongolian language model by combining language characteristics and trigger. Structured language model[3], as the model combined language model and inference network model, can support structured query as inference network model as well as boast flexible smooth technology like language model

This paper improves retrieval performance by studying the selection of retrieval unit of the Mongolian informational retrieval system. Experiments are conducted for each retrieval unit through the lemur retrieval platform. Thereby, experimental results are compared to find the most suitable election method of retrieval units.

Information Retrieval Model and Tool

Information Retrieval Model. Information retrieval models are mainly divided into the following categories, namely, Boolean Model, Vector Space Model, Probability Model and language model, etc. According to the characteristics of Mongolian language, vector space model and language model are chosen. Vector space model introduces the concept of space into the model by regarding documents and query as the vectors of dimensional vector space. With the angle between vectors as the measurement, it calculates the similarity of documents and query. The language model is developed recently. Language model is the best model to reflect the inherent law of the language[4].

(1) TF-IDF Model. TF-IDF (term frequency–inverse document frequency) is a common weighting technique used for data retrieval and information exploration. As a statistical method, TF-IDF can be used to estimate the importance of a term on a file collection or one file of a corpus. The importance of terms proportionally increases to its frequency in file. However, at the same time, it proportionally decreases to its frequency in file. In other term, if the term w appears in file d frequently but rarely in other files, then the term w boasts good ability to distinguish and suitable to distinguish the article d and others.

In a given file, term frequency (TF) is defined as the frequency of a given term in the file. The same term (whether the term is important or not) may have higher frequency in longer file than shorter one. To prevent deviating to longer file, the number will be normalized generally.

The TF (Term Frequency) of the term, w, in the file d, is ratio of the count of w in d (w, d) and the that of w in the file collection D:

tf (w,d)= count (w, d) / count (w, D)

= count (w, d) / sum { i = 1..n | count (w, d[i])

Inverse document frequency (IDF) is a measurement of a term's general importance. For the IDF of a given term can be reached by the total number of files dividing the number of term-included files and then getting the logarithmic of the result. IDF (Inverse Document Frequency) of w in the whole file collection is the logarithm of the radio of the total file number, n, and w-included file number, docs (w, D).

idf = log (n / docs (w, D))

tf-idf model calculates the weight of query string, q, consisting of every file d and key terms w[1]…w[k] according to tf and idf. It is used to describe the match degree of q and d:

tf-idf (q, d) = sum { i = 1..k | tf-idf (w[i], d) }

= sum { i = 1..k | tf (w[i], d) * idf (w[i]) }

(2) Vector Space Model. Vector Space Model (VSM) was put forward by alton et al. in 1960s and successfully applied to the famous SMART system. The following decades since then, the model was widely applied in many fields, including text classification, automotive retrieval and information retrieval.

Vector Space Model regards user query and documents as n vectors that are mutually independent. It is presumed that D refers to document collection. The vector n in the document is expressed as t1,t2,⋯,tn. di refers to a document of the collection D. The weight of t1,t2,⋯,tn in the di is expressed as wi1,wi2,⋯,win, i.e., di=(wi1,wi2,⋯,win). The larger the value of w is, the more important t is in the document and the better t reflects di. The smaller the value of w is, the less important t is in the document and the worse t reflects di. The range of w is [0,1]. q=(w1,w2,⋯,wn) refers to user query and similarity is used to reflect the relationship between user query and documents. The equation is as follows:

$$\cos(D_i, Q) = \frac{\sum_{k=1}^{n} w_{ik} \times w_k}{\sqrt{\sum_{j=1}^{n} w_{ij}^2 \sum_{j=1}^{n} w_j^2}} \tag{1-1}$$

From the equation, the similarity is calculated through the cosine of document and query vectors. In the equation, w refers to the weight of terms. Its value is proportional to the frequency of the term in the document and inversely proportional to the number of term-included documents in D. The w formula is as follows:

$$w_{ik} = \frac{tf_{ik} \times \lg(N / n_k)}{\sqrt{\sum_{j=1}^{N} (tf_{ij} \times \lg(N / n_j))^2}} \tag{1-2}$$

In the equation, tf_{ik} means the frequency of t_k in d_i. N refers to the sum of documents in the collection and n_k means the frequency of the term t_k in the document.

The advantages of Vector Space Model include: (1) Weight distribution is made to terms. The value of weight is used to express the similarity between the term and document, which is better than Boolean Model; (2) Documents with similar contents are classified based on the calculated similarity; (3) Vector Space Model can better get close the user's intention than Boolean Model.

Vector Space Model has some disadvantages: (1) Similarity calculation is quite complicated which can affect retrieval speed; (2) It is difficult to determine the weight of index terms; (3) The independent hypothesis of index terms does not conform to the reality[6].

(3) Language Model. Language model is the mathematical model describing the inherent laws of statistics and structure of natural language. In terms of natural language processing, computer's understanding of natural language is based on language model. Therefore, the language model is widely applied in natural language processing. For instance, in the field of voice recognition, computer has to complete two things to realize the conversion from voice to text: one is to recognize the voice and two is to identify the corresponding text of the voice. At the time, language model should be employed to give scores of all candidates [7]. Language Model is also widely applied in machine translation, handwriting character recognition and text retrieval.

In the letter-based n-gram language model, C represents n letters ordered in the term, i.e., C=c1,c2,…cn. N-gram model holds the probability of any letter ci is only related to n-1 previous letters. It is expressed by the following equation[25]:

$$P(C)=P(c_1,c_2,\cdots,c_n)=P(c_1)P(c_2\mid c_1)\cdots P(c_i\mid c_{i-n+1}\cdots c_{i-1})\cdots P(c_n\mid c_1\cdots c_{n-1}) \quad (1\text{-}3)$$

The following equation is further streamlined:

$$\log P(c_1,c_2,\cdots c_n)=\sum_{i=1}^{n}\log p(c_i\mid c_{i-n+1}\cdots c_{i-1}) \quad (1\text{-}4)$$

For the model parameter training, Maximum Likelihood Estimation is employed to calculate the probability of every parameter. In the experiment, we give n the value from 2 to 5.

If n=2, then the equation of conditional probability of the bigram model is as follows:

$$P(c_n\mid c_{n-1})=P(c_{n-1}c_n)/P(c_{n-1}) \quad (1\text{-}5)$$

If n=3, then the equation of conditional probability of the trigram model is as follows:

$$P(c_n\mid c_{n-2}c_{n-1})=P(c_{n-2}c_{n-1}c_n)/P(c_{n-2}c_{n-1}) \quad (1\text{-}6)$$

The equation of conditional probability of the n-gram model (n=4) is as follows:

$$P(c_n\mid c_{n-3}c_{n-2}c_{n-1})=P(c_{n-3}c_{n-2}c_{n-1}c_n)/P(c_{n-3}c_{n-2}c_{n-1}) \quad (1\text{-}7)$$

The equation of conditional probability of the n-gram model (n=5) is as follows:

$$P(c_n\mid c_{n-4}c_{n-3}c_{n-2}c_{n-1})=P(c_{n-4}c_{n-3}c_{n-2}c_{n-1}c_n)/P(c_{n-4}c_{n-3}c_{n-2}c_{n-1}) \quad (1\text{-}8)$$

With the trigram model as the example, the probability of the letter in the trigram model is only related to the previous two letters. From the equation, it can be seen that both P(cn-2cn-1cn) and P(cn-2cn-1)are unknown. In order to get P(cn-2cn-1cn) and P(cn-2cn-1), C(cn-2cn-1cn) and C(cn-2cn-1) can be reached statistically, which represent the frequency of (cn-2cn-1cn) and (cn-2cn-1). Based on the maximum likelihood estimation method, it reaches:

$$P(c_{n-2}c_{n-1}c_n)=\frac{C(c_{n-2}c_{n-1}c_n)}{\sum_{(c_{n-2}c_{n-1}c_n)}C(c_{n-2}c_{n-1}c_n)} \quad (1\text{-}9)$$

In the equation, $\sum_{(c_{n-2}c_{n-1}c_n)}C(c_{n-2}c_{n-1}c_n)$ refers to the total number of trigram. With the same method, it reaches:

$$P(c_{n-2}c_{n-1})=\frac{C(c_{n-2}c_{n-1})}{\sum_{(c_{n-2}c_{n-1})}C(c_{n-2}c_{n-1})} \quad (1\text{-}10)$$

In the equation, $\sum_{(c_{n-2}c_{n-1})}C(c_{n-2}c_{n-1})$ refers to the total number of bigram.

It is presumed that the term length is L, L-1, the total number of bigram, can be got easily. The total number of trigram is L-2. Similarly, L-3, the total number of n-gram (n=4) and L-4, the total number of n-gram (n=5) are reached. For trigram, the equation is as follows:

$$P(c_n \mid c_{n-2}c_{n-1}) = \frac{P(c_{n-2}c_{n-1}c_n)}{P(c_{n-2}c_{n-1})} = \frac{C(c_{n-2}c_{n-1}c_n)/(L-2)}{C(c_{n-2}c_{n-1})/(L-1)} \tag{1-11}$$

We calculate $C(c_{n-2}c_{n-1}c_n)$ and $C(c_{n-2}c_{n-1})$ for each term in the training corpus and get L, the length of the term.

Introduction of Information Retrieval Tool and Method. Currently, most people use information retrieval tool to carry out experiments. Of them, the most popular tools include Lucene and Lemur. Then, it introduces Lemur, the retrieval tool employed in this experiment.

Lemur system, launched jointly by CMU and UMass, is used for natural language model and information retrieval researches. Based on the system, it realizes the natural language model, traditional vector space model and Okapi's ad hoc or distributed retrieval. Here, structured query, cross-language equation, filtering and clustering are carried out. The key steps of retrieval with Lemur system include: analytical query. Doc analysis can be employed to produce relevant documents of inverted index; retrieval with RetEval and result evaluation.

Retrieval Unit Selection Methods

The retrieval unit selection method in the Mongolian information retrieval system is the importance indicator influencing the performance of information retrieval and one of key issues of Mongolian information retrieval. It is the focus of the research. At first, according to characteristics of Mongolian language, Mongolian language consists of root suffix and several different affixes and configurations. Therefore, we decide to use term and root + affix as retrieval unit. In the experiment, root + affix as retrieval unit are divided into: root + 1 affix, root + 2 affixes; root + 3 affixes. First, the reason why 3 affixes are chosen is that root covers the basic meaning of terms for Mongolian words. Affixes have different types, e.g., word-formation, configuration and suffix. Generally, affix following root is word-formation affixes, whose contribution to terms cannot be ignored. Then, it may also be configuration affix and suffix. Suffixes have no practical significances. Therefore, its contribution to terms can be ignorable. Before we carry out the experiment, terms segmentation is carried out for original corpus. For 276143 repeated Mongolian words, the number of 3 kinds (root + 1 affix, root + 2 affixes; root + 3 affixes) of words is 212756. Therefore, both theoretical and experimental results show out retrieval unit is selected reasonable.

Experiment and Analysis

According to the characteristics of Mongolian language, we classify retrieval units as term form and root+affix. The term form is used as the baseline of the experiment. Root + affix retrieval unit are divided into: root + 1 affix, root + 2 affixes; root + 3 affixes.

It should be noted that if affixes behind roots fail to satisfy the experiment needs, then it is replaced with the room. For example, the experiment needs to employ root+1 affix. If the word is the root of the term, then the root replaces the term. The experiment needs to employ root+2 affixes, when the number of the affix is not enough, then it is replaced with root + 1 affix and so on.

We use TF-IDF model, Vector Space Model and Language Model in the experiment on retrieval unit selection. The following results are shown in Tab. 1 and Fig. 1

Table 1 Comparison Table for Retrieval Units in tf-idf Model

	Text Form	Root Form	Root + 1 affix	Root + 2 affixes	Root + 3 affixes
0.0	0.8406	0.9306	0.9106	0.9406	0.9211
0.1	0.7646	0.8646	0.7646	0.8646	0.7566
0.2	0.7002	0.7667	0.7167	0.7812	0.7043
0.3	0.6523	0.7092	0.6092	0.7172	0.6123
0.4	0.6174	0.5421	0.5421	0.6421	0.5367
0.5	0.4753	0.4912	0.4912	0.5912	0.4567
0.6	0.4253	0.3971	0.4571	0.5561	0.4032
0.7	0.3985	0.3724	0.4724	0.4824	0.4235
0.8	0.2754	0.2004	0.3004	0.3234	0.2789
0.9	0.1523	0.1293	0.2293	0.2493	0.2256
1.0	0.1252	0.0447	0.1447	0.1215	0.1237
AVG	0.4933	0.4953	0.5125	0.5699	0.4947

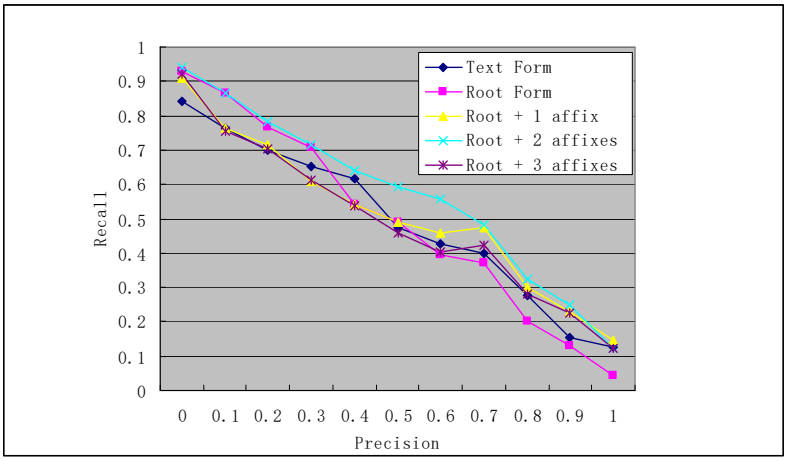

Figure 1 Comparison Figure for Retrieval Units in tf-idf Model

The experiment results show that in the TF-IDF model, the better retrieval performance is reached by the retrieval unit of root + 2 affixes. The mean value is up to 0.5699.

Retrieval with Vector Space Model, we get the results as Tab. 2 and Fig. 2 show.

Table 2 Comparison Table for Retrieval Units in Vector Space Model

	Text Form	Root Form	Root + 1 affix,	Root + 2 affixes	Root + 3 affixes
0.0	0.8226	0.9226	0.9226	0.9226	0.8906
0.1	0.7819	0.8619	0.7619	0.8819	0.7519
0.2	0.7012	0.7807	0.7807	0.7307	0.7503
0.3	0.6523	0.7207	0.6207	0.7201	0.6645
0.4	0.6027	0.6051	0.605	0.7051	0.6123
0.5	0.5441	0.5441	0.5441	0.6441	0.5468
0.6	0.5897	0.5192	0.4998	0.6096	0.4784
0.7	0.5287	0.3187	0.4187	0.5485	0.4213
0.8	0.3975	0.2975	0.3975	0.4875	0.3674
0.9	0.2607	0.1603	0.26	0.2807	0.3211
1.0	0.1626	0.0636	0.1636	0.1826	0.1333
AVG	0.5494	0.5267	0.5431	0.6103	0.5398

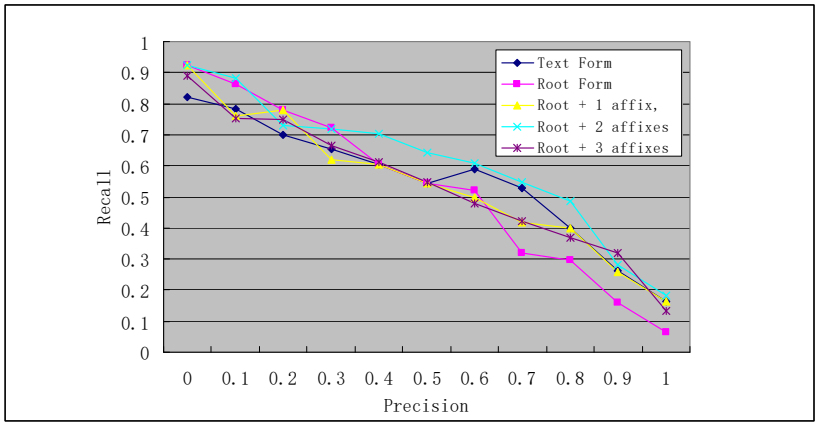

Figure 2 Comparison Figure for Retrieval Units in Vector Space Model

The experiment results show that in the Vector Space Model, the better retrieval performance is reached by the retrieval unit of root + 2 affixes. The mean value is up to 0.6103.

Retrieval with Lemur Language Model, we get the results as Tab. 3 and Fig. 3 show

Table 3 Comparison Table for Retrieval Units in Lemur Language Model

	Text Form	Root Form	Root + 1 affix,	Root + 2 affixes	Root + 3 affixes
0.0	0.8261	0.9161	0.9067	0.9264	0.8865
0.1	0.7965	0.8255	0.8155	0.8765	0.8355
0.2	0.7441	0.7991	0.7991	0.8241	0.8052
0.3	0.7002	0.7702	0.7702	0.7702	0.7632
0.4	0.6709	0.6201	0.6207	0.7309	0.6007
0.5	0.6502	0.5902	0.5902	0.6712	0.5302
0.6	0.5458	0.5158	0.4958	0.6158	0.4356
0.7	0.4925	0.3121	0.412	0.5234	0.4111
0.8	0.3472	0.2952	0.3952	0.4572	0.3852
0.9	0.2452	0.1422	0.2422	0.3521	0.2345
1.0	0.1581	0.0481	0.1123	0.1981	0.1268
AVG	0.5615	0.5304	0.5599	0.6314	0.5467

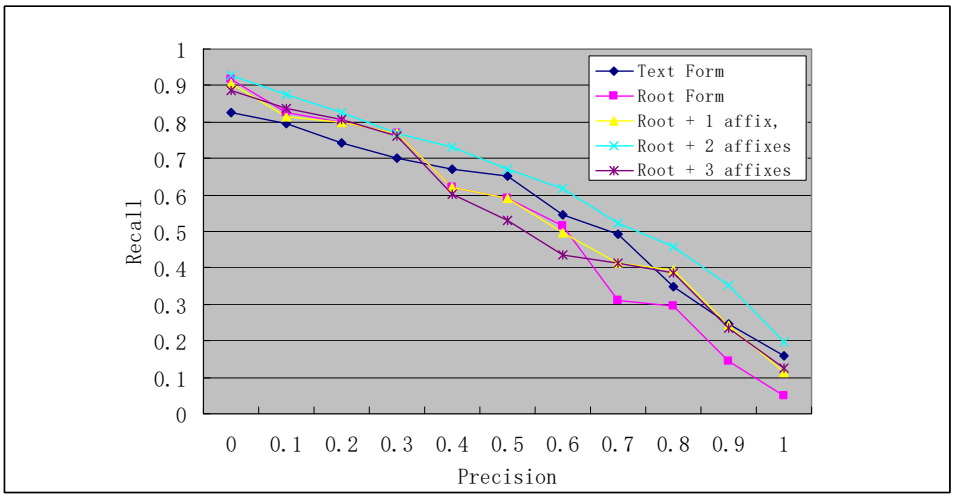

Figure 3 Comparison Figure for Retrieval Units in Lemur Language Model

The experiment results show that the better retrieval performance is reached by the retrieval unit of root + 2 affixes. The mean value is up to 0.6314.

Experiment summary: among three retrieval models (TF-IDF model, vector space model, language model), the systematic retrieval performance of root + affix form (Text, root, root + 1 affix, root + 2 affixes; root + 3 affixes.) in using language model than the other two models. Root + 2 affixes reach the best retrieval performance of root + affix form. The even value is up to 0.6314.

Conclusions

This paper studies the selection method of retrieval units of Mongolian information retrieval system. The selection of retrieval directly affects the retrieval effects and performance. Proper retrieval unit can greatly improve the retrieval efficiency of Mongolian information. According to the characteristics of Mongolian language, retrieval units can be divided into: text and root + affix forms. The root + affix form can be further divided into stem + 1 affix, stem + 2 affixes, stem + 3 affixes. Lemur system, as the Mongolian information retrieval platform, is used for experiment: Lemur is the open-source information retrieval platform. Three kinds of information retrieval models are employed in the experiment, respectively, TF-IDF Model, vector space model, language model to explore the retrieval effect on text, root + 1 affix, root + 2 affixes; root + 3 affixes. It is found that the best performance is achieved by root + 2 affixes under the Lemur language model, with the even value up to 0.6314.

This paper carries out relevant researches on key issues of Mongolian information retrieval. However, further study and improvements should be made for some questions and works in the future study. According to the characteristics of Mongolian language, Mongolian words consist of root and multiple suffixes. But in this paper, consideration is only given to the condition of root and 3 affixes. Affixes should be increased in the experiments of future researches.

Acknowledgements

Fund project: Ministry of Information Industry Projects ([2011]506)

References

[1] Zhang Guoqiang, The Research on Constructing Layered Mongolian Language Model. Hohhot: Inner Mongolia University, 2008.

[2] Liu Zhiwen, Researching of Long-Distance Mongolia Language Model Based on Trigger Pair .Hohhot: Inner Mongolia University, 2008.

[3] Jin Wei, Research of Mongolian Information Retrieval Model. Hohhot: Inner Mongolia University, 2009

[4] Ma Shaoping, Zhang Min, Information Retrieval: Over the past three decades, how long we have gone far and what forefront progress has been made in Chinese information processing, Twenty-fifth Anniversary Academic Conference Proceedings of Chinese Information Society

[5]G.Salton,M.E.Lesk. Computer Evaluation of Indexing and Text Processing. Journal of the ACM.1968,15(1):8-36

[6] Lou Luqun et al. Research of Language Model in Information Retrieval. Computer Engineering, 2007

[7] Wu Genqing. Research and Application of Statistical Language Model, Beijing: Tsinghua University, 2004

[8] Xu Zhiming, Wang Xiaolong, Guan Yi, Data Smoothing Techniques of N-gram Language Model, Application Research of Computers, 1999, (7): 37 to 41.

[9] Church,A Comparison of the enhanced Good-Turing and deleted estimation methods for estimating probabilities of English bigrams.,Computer Speech and language ,1991,5(1):19~54.