2010 2nd International Conference on Information and Multimedia Technology (ICIMT 2010)
IPCSIT vol. 42 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V42.7

Personal Multimedia Data Retrieval Query Expansion and Similarity Algorithm Improvement based WordNet

Tiantan Han[1], Wendong Wang[1], Xiangyang Gong[1], Jian Ma[2], Canfeng Chen[2], Xiaogang Yang[2]

[1]State Key Lab of Switching & Networking Technology,Beijing University of Posts and Telecommunications, Beijing

[2]Nokia Research Center,Beijing

tiantan1220@163.com, wdwang@bupt.edu.cn,xygong@bupt.edu.cn, Jian.J.Ma@nokia.com, canfeng-david.chen@nokia.com, ext-xiaogang.1.yang@nokia.com

Abstract-Because the probability that people use the same form of the words to describe the same semantic concept is very small especially for the different kind of Medias, we need query expansion when searching for the short texts which represent multimedia information. In this paper we calculate words similarity based the WordNet Synonym to tell us which synonyms are more common and more related, and then we can ascertain how many synonyms we expand and the weight of the Synonyms. Then more related text would be searched. And we optimize similarity algorithm based the similarity between synonyms to help us to search and sort the texts based the meaning of keywords not only the frequency of the keywords.

Keywords-WordNet; Information retrieval; Query expansion; Semantic similarity; Text similarity; Clucene

1 Introduction

People often have a lot of Personal data on the network, and sometimes they need to retrieve all of them, for example, when seeing a picture, we usually want to know when, where, who the picture related. so we need all of the data related to the picture, and when retrieving and managing personal data in the network, we still mainly use keyword matching, for example, when retrieving video, photos and other multimedia information, we still rely on tag matching, so we get tags, time, and all text information of the multimedia, and make them to be a text, but if using these tags to indicate the multimedia, we can only get a short text, and other personal data are always short text too, such as email, blog, SMS. All these personal data can be searched as a short text, so when we make all of personal data as text and retrieving them, we can get all kinds of personal data, including multimedia information and short texts. And all of them are related.

But when searching for these short texts, we will get little short text related. Because the probability that people use the same form of the words to describe the same semantic concept is very small, especially for the different kind of Medias. So we need query expansion when searching for the multimedia information. But how many synonyms we expand and the weight of the Synonym are difficult to ascertain. So calculating words similarity based the WordNet Synonym sets are necessary and then using it in query expansion to get more related information to the query. If the similarity between synonyms of WordNet is calculated, we can set a threshold to decide how many synonyms we expand and make the similarity as the weight of the synonyms we expand. On the results of the expanded short text searching, optimizing the similarity algorithm to calculate a more accurate short text similarity to get a more reasonable ranking is necessary too. If related texts are found out based synonyms expansion, we can optimize similarity algorithm based the similarity between synonyms.

This paper presents a method for calculating word similarity in the WordNet synonym set and the corresponding optimization method of similarity algorithms and test the result based Clucene information retrieval engine to show the excellence of the optimization similarity algorithm. The similarity between synonyms calculated based this method tell us which synonyms is more common and more related. And through the test in Clucene we find that the optimized similarity algorithms can find out the texts including the meaning of keywords not only the texts including just the keywords, and the sort of searching result is based how much the text contain the meaning of the keywords not only the frequency of the keywords.

The 2th part is related work and overview. The 3th part is Information retrieval query expansion by WordNet. The 4th part is Optimize text similarity measure corresponding query expansion. The 5th part is Query Result Test

2 Related Work & Overview

2.1 information retrieval query expansion

What often happened in information retrieval is: the key word that the user inputted has the same meaning of the word that appears in the text, but it is a different word. synonym word but diverse, leads to associated text can not be retrieved. To solve the problem which is caused by the diverse of synonymous words, the query expansion is necessary. For example, expanding the query "abstract" into "abstract / outline / nonobjective", not only can effectively address mismatch caused by diverse of synonymous words but also can retrieve more relevant documents.

There are many methods of query expansion, and two methods used commonly is : (1) analysis the first search result and add information which is extracted from feedback into query; (2) use some resources to extend the query directly. The effect of method (1) is unstable [2]. The effect strongly depends on the results of the first time and the accuracy of user's feedback, especially for the search of short term and short target text. The information that abstracted from the feedback text is limited and the first search result is unstable. Method (2) need to use some kind of resources which containing words relation information. WordNet [7] is one of such resource, which provides the complex relationship between the English words, including synonyms, antonym, qualifier and other relevant information [3].

In this paper, we will focus on and discuss the use of WordNet synonym sets in English information retrieval query expansion, and calculate the synonym similarity between the words by other factors in the synonym set, the similarity of words are based on degree of replaceable and the using frequency of the word. And use the synonym similarity to ascertain how many synonyms we expand and the weight of the Synonym.

2.2 Text similarity algorithm & Clucene

Text Similarity Computing has wide application in information retrieval [4], data mining, machine translation, document copy detection [11] etc... Text similarity calculation method is generally divided into two categories: similarity calculation based on statistical methods and understanding based on semantic similarity measure.

In Vector space model [5], the text was regarded as a vector space which formed by key words, query terms are also converted into a vector and the cosine angle between two vectors is used to identify their relevance. If the query is expressed as $q = (q_1, q_2, q_3 ...)$, text is represented as $d = (d_1, d_2, d_3 ...)$ then the similarity between the two is: $similarity(q,d) = \frac{\sum_{i=1}^{n}(q_i d_i)}{\sqrt{\sum_{i=1}^{n}(d_i)^2 \sum_{i=1}^{n}(q_i)^2}}$ Clucene is based Vector space model, and its sorting algorithm is [6]:

$$score(q,d) = coord(q,d) * queryNorm(q)$$
$$* \sum_{tinq} (tf(tind) * idf(t)^2 * t.getBoost() * norm(t,d))$$

In this paper, we optimize the sorting algorithm based Clucene; we use similarity between the synonyms word to merge expanding words to reduce the dimension of vector space model. So that we import similarity preliminary calculation based on word semantic understanding into similarity calculation based on statistics.

2.3 Overview of WordNet

WordNet is developed in the Princeton University Cognitive Science Laboratory under the guidance of Prof. G. Miller [1]. It is applied to many areas related to semantic analysis in natural language processing. At present, WordNet has become a de facto international standard, rationality of WordNet framework has been recognized by lexical semantics community and computing dictionary community.

WordNet organize lexical information based word meaning rather than form. WordNet synonym sets (Synset) is on behalf of the Concept. WordNet organized the English vocabulary as a synonym collection (Synset) [9], a Synset indicate a concept; and create variety semantic relations [10] between concepts, such as the upper and lower, antisense and synonyms. This constitutes a relatively complete system of lexical semantic networks. Through this process, the original abstract concept will be formalized, specific and could be operated by the meaning of vocabulary. Various semantic links and reasoning can be established between concepts [8].

In this paper, by the analysis of other factors in WordNet Synset, we calculate each word similarity between words in the same Synset, the similarity of words based on Degree of replacement and using frequency. We use synonyms sequence ranked by synonymous similarity to expand query, and we use synonymous similarity to merge synonyms and calculate word frequency in vector space model by Clucene.

3 Information retrieval query expansion by WordNet

WordNet represent semantic concept by Synset, to each Synset, there are several lines to represent, as following lines:

$$s(synset\_ID, w\_num, 'word', ss\_type, sense\_number, tag\_count)$$

$synset\_ID$ is unique identifier of this Synset. w_num is the order of the word in the Synset. $'word'$ Identify this term is in the Synset. ss_type Identify Synset type; type is limited in nouns, verbs, adjectives, adverbs. $sense\_number$ Identify frequency of the word, the greater the value, the smaller the frequency. tag_count Identify the frequency of the word in a corpus. When we calculate the similarity in a Synset, we mainly use $sense\_number$ parameter.

Many words appear in some different Synsets, because the term has more meaning, and therefore provided by WordNet Synsets, for a term b, we may get m Synsets. As follows:

$b(k_1) \to b_{11}(k_{11}), b_{12}(k_{12}), b_{13}(k_{13}), \cdots b_{1n_1}(k_{1n_1}))$ $\quad$ $b(k_2) \to b_{21}(k_{21}), b_{22}(k_{22}), b_{23}(k_{23}), \cdots b_{2n_2}(k_{2n_2}))$

$b(k_3) \to b_{31}(k_{31}), b_{32}(k_{32}), b_{33}(k_{33}), \cdots b_{3n_3}(k_{3n_3}))\ldots$

$b(k_m) \to b_{m1}(k_{m1}), b_{m2}(k_{m2}), b_{m3}(k_{m3}), \cdots b_{mn_m}(k_{mn_m}))$ This means the word b is in m Synsets, $k_1 \cdots k_m$ is the $sense\_number$ value that b has in each Synset, and in c-th Synset, the number of synonyms is n_c, $k_{c1} \cdots k_{cn_c}$ are the synonyms $sense\_number$ value in c-th Synset.

More research is calculating the similarity of Synsets and the relativity between different Synsets, little research is calculating the similarity between synonyms. So we use the following formula calculate similarity between word b and synonyms b_{ij}:

$$sim(b, b_{ij}) = a\frac{1}{k_i} + b\frac{1}{k_{ij}} + c\frac{n_i}{\sum_{p=1}^{n_i} k_{ip}} + d\frac{n_i}{\sum_{q=1}^{m}\sum_{p=1}^{n_q} k_{qp}} \quad \begin{matrix} a+b+c+d=1 \\ 0<a,b,c,d<1 \end{matrix}$$

4 Optimize text similarity measure corresponding query expansion

Now, we optimze text similarity algorithm based Clucene. In the third part we expand the query, that means we get more related text through different words which has the similar meaning, so when we are sorting, we should polymerize the words to indicate the degree of meaning in the text.

Original vector $c = (c_1, c_2, \cdots c_m)$.

Original query vector $u = (u_1, u_2, \cdots u_m)$.

Original text vector $t = (t_1, t_2, \cdots t_m)$

After query expansion:

Expand vector: $v = (v_1, v_2, v_3 \cdots v_n)$

Expand query vector: $q = (q_1, q_2, q_3 \cdots q_n)$

Expand text vector $d = (d_1, d_2, d_3 \cdots d_n)$

$q_1, q_2, q_3 \cdots q_n, d_1, d_2, d_3 \cdots d_n$ is frequency.

$v_1, v_2, v_3 \cdots v_n$ is word.

if(v_j is expanded by v_i)

$q_j = q_i$

end if

for($v_i, v_j, i = 1,2,3 \cdots n, j = 1,2,3 \cdots n$)

if(v_j is expanded by v_i)

merge $v_i, v_j tov_{ij}$ v_i

merge $q_i, q_j toq_{ij}$

merge $d_i, d_j tod_{ij}$

$q_{ij} = (q_i + q_j)/2$

$d_{ij} = d_i + d_j * sim(v_i, v_j)$

end if

end for

Through the above calculation, the vector is still the m-dimensional vector, but the matched texts include the search results with extended terms, and the word frequency of text vector increased. In the process we account word similarity factor, the original word frequency and expansion word frequency. This can make query results have more relevant text, and make text similarity more precise.

5 Query Result Test in Clucene

The Example of Synonym expansion similarity calculation and ranking result is as follows.

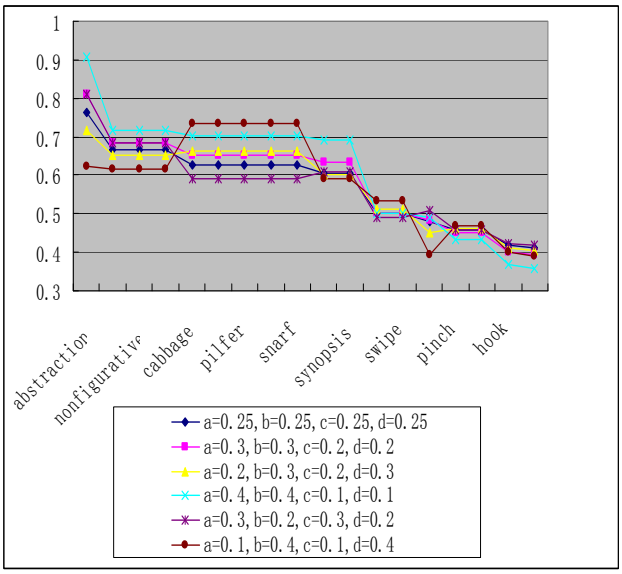
First we test the similarity between synonyms. Take “abstract” and check as an example.

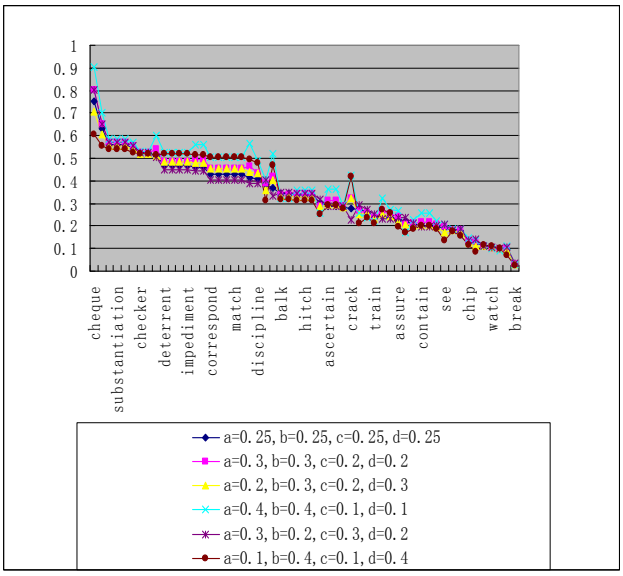
“abstract” synonyms are: abstraction, abstractionist, nonfigurative, nonobjective, cabbage, filch, pilfer, purloin, snarf, précis, synopsis, nobble, swipe, outline, pinch, sneak, hook, lift.

“check” synonyms are: cheque, assay, confirmation, substantiation, verification, checkout, chequer, tab, deterrent, hinderance, hindrance, impediment, halt, correspond, gibe, jibe, match, tally, insure, discipline, chit, curb, balk, baulk, arrest, hitch, stay, handicap, ascertain, ensure, condition, crack, stop, agree, train, control, bridle, assure, fit, tick, contain, moderate, retard, see, chink, delay, chip, hold, learn, watch, determine, mark, break.

The next two maps are the similarity of these synonyms. X-axis is the synonyms, Y-axis is the similarity.

Each curve is a combination value of a, b, c and d.





Obviously, after the similarity calculation and sorting, the synonymous sequence meet our daily usage habits more and we can set a value for more precise synonym expansion. We can accord our need to determine the value of a, b, c and d.

In the next test, we use the similarity between synonyms when $a = b = c = d = 0.25$ to see the result of text similarity algorithm optimization and ranking result.

Now we use Clucene as our search engine for test the search result which is based on synonym expansion and improvement of text similarity, we compare them to illuminate the benefit of expansion and improvement of text similarity. Input "abstract" and "check" as query word. The synonym of "abstract" appeared in the texts are "abstraction" and "snarf", and $sim(abstract, abstration) > sim(abstract, snarf)$ .the synonym of "check" appeared in the texts are "assay" and "checkout" and $sim(check, assay) > sim(check, checkout)$.

We get six docs from web, they excerpt from the following URL.

Doc1 (140 words. include: abstract(1),check(1))
From:http://www.mto.gov.on.ca/english/dandv/driver/record.shtml

Doc2(163 words. include: assay(1),check(1),abstraction(1),abstract(1))
From:http://www.ncbi.nlm.nih.gov/bookshelf/br.fcgi?book=erta160&part=A257796

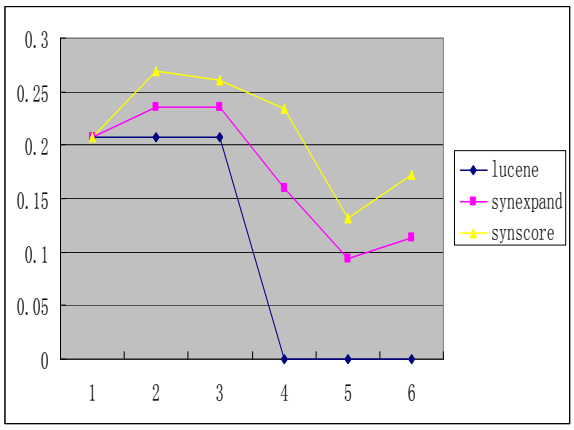
Doc3(171 words. Include: checkout(1),check(1),snarf(1),abstract(1))
From:http://ftp.cs.duke.edu/courses/spring06/cps100/assign/jotto/

Doc4(153 words. Include: checkout(1),assay(1),snarf(1),abstraction(1))
From:http://scienceblogs.com/insolence/2007/04/fear_of_the_frame.php

Doc5(166 words. Include: checkout(1),snarf(1))
From:http://www.wired.com/underwire/2010/06/can-thundercats-anime-makeover-rehabilitate-snarf/

Doc6(144 words. Include: abstraction(1),assay(1))
From:http://pubs.acs.org/doi/abs/10.1021/jf052731u

The chart is the score of the six docs in different way. X-axis is the docs, Y-axis is the score. The blue line is the score based the original Clucene search and similarity calculation method, there is no query expansion, so we can only get doc 1, 2, 3, and the score is same. The red line is the score based extension, but as a synonym the similarity joins a fixed value 0.3, so we can get all of the docs and doc 2, 3 get a higher score than doc 1. The yellow line is the score based the use of query expansion and similarity algorithm described in this paper.



Obviously, we not only get with the text including 'abstract' and 'check', but also search out the text with their synonyms, and the similarity of the synonym make the text score more exact: make more high similarity, and the degree that synonyms impact the text similarity is decided by the similarity of synonyms.

Now we explain why the second doc get higher score than the first doc in yellow line, that is because $sim(check, assay)$ and $sim(abstract, abstraction)$ is big enough, so the meaning of check and abstract in doc 2 is more than in doc 1. So if query is check and abstract, doc 2 match more meaning. Doc 4 is higher than doc 1 is because checkout and assay include more "check" meaning than check. And snarf and abstraction include more "abstract" meaning too.

6 Conclusions and future work

This paper calculate and quantify the synonyms similarity through WordNet Synset so we get more precise query expansion, and improve the quality of query expansion, and put forward for This query expansion method we improved the similarity calculation for search results, so we get a more exact text similarity.

For the next step we need to: (1) further improve the calculation of synonyms similarity, so that it would more line usage especially the value of a b c d; (2) use HowNet to calculate similarity in Chinese information for Chinese Query Expansion; (3) further optimize text similarity algorithm for query expansion, making expanded text similarity calculation more accurate.

7 Acknowledgements

This work is supported by Nokia Research Center Beijing, EU FP7 Project EFIPSANS (INFSO-ICT-215549), the National 863 project Grant No. 2009AA01Z210 and No. 2009AA01Z250.

8 References

[1] G. A. Miller.: WordNet: a lexical database for English. Comm. ACM, vol.38, No.11, pp. 39-41, 1995.

[2] Ellen M. Voorhees.: Query expansion using lexical-semantic relations. In proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland. 61-69. 1994.

[3] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller.: Five papers on WordNet. CSL Report 43, Cognitive Science Laboratory, Princeton University. 1990.

[4] R. K. Srihari, Z. F. Zhang, and A.B.Rao.: intelligent indexing and semantic retrieval of multimodal documents. Information Retrieval, vol.2, pp.245-275, 2000.

[5] G. Salton and M. J. McGill.: Introduction to Modem Information Retrieval. McGraw-Hill. 1983

[6] Otis Gospodnetic, Erik Hatcher.: Lucene in Action.

[7] Shuang Liu, Fang Liu, Clement Yu, Weiyi Meng.: An Effective Approach to Document Retrieval via Utilizing WordNet and Recognizing Phrases. In proceedings of the 27th Annual International ACM SIGIR Conference on Research and development in Information Retrieval, Sheffield, Yorkshire, UK. 2004

[8] Qiu. Y, Frei. H. P.: Concept based query expansion. In Proceedings of the 16th annual international ACM SIGIR

conference on Research and development in information retrieval. ACM Press, Pittsburgh, Pennsylvania, USA. 160-160. 1993

[9] Giannis Varelas, Epimenidis Voutsakis, Paraskevi Raftopoulou.: Semantic Similarity Methods in WordNet and their Application to Information Retrieval on the web. 7th ACM International Workshop on Web Information and Data Management, Bremen, Germany. 2005.

[10] G. A. Miller, Claudia Leacock, Randee Tengi, Ross. T. Bunker.: A Semantic Concordance. Proceedings of the 3rd DARPA Workshop on Human Language Technology. 1993

[11] A. Budanitsky and G. Hirst.: "Semantic distance in WordNet: an experimental, application-oriented evaluation of five measures". Proceeding of Workshop WordNet and Other Lexical Resources, Second Meeting North Am. Chapter Assoc. for Computational Linguistics, 2001.