How to Get the Same News from Different Language News Papers

T. Pattabhi R. K Rao
AU-KBC Research Centre
Anna University Chennai

Sobha Lalitha Devi
AU-KBC Research Centre
Anna University Chennai
sobha@au-kbc.org

Abstract

This paper presents an ongoing work on identifying similarity between documents across News papers in different languages. Our aim is to identify similar documents for a given News or event as a query, across languages and make cross lingual search more accurate and easy. For example given an event or News in English, all the English news documents related to the query are retrieved as well as in other languages such as Hindi, Bengali, Tamil, Telugu, Malayalam, Spanish. We use Vector Space Model, a known method for similarity calculation, but the novelty is in identification of terms for VSM calculation. Here a robust translation system is not used for translating the documents. The system is working with good recall and precision.

1 Introduction

In this paper we present a novel method for identifying similar News documents from various language families such as Indo-European, Indo- Aryan and Dravidian. The languages considered from the above language families are English, Hindi, Bengali, Tamil, Telugu, Malayalam and Spanish. The News documents in various languages are obtained using a crawler. The documents are represented as vector of terms.

Given a query in any of the language mentioned above, the documents relevant to the query are retrieved. The first two document retrieved in the language of the query is taken as base for the identification of similar documents. The documents are converted into terms and the terms are translated to other languages using bilingual dictionaries. The terms thus obtained is used for similarity calculation. The paper is further organized as follows. In the following section 2, related work is described. In section 3, the algorithm is discussed. Section 4 describes experiments and results. The paper concludes with section 5.

2 Related Work

In the past decade there has been significant amount of work done on finding similarity of documents and organizing the documents according to their content. Similarity of documents are identified using different methods such as Self-Organizing Maps (SOMs) (Kohonen et al, 2000; Rauber, 1999), based on Ontologies and taxanomy (Gruber, 1993; Resnik, 1995), Vector Space Model (VSM) with similarity measures like Dice similarity, Jaccard’s similarity, cosine similarity (Salton, 1989).

Many similarity measures were developed, such as information content (Resnik, 1995) mutual information (Hindle, 1990), Dice coefficient (Frakes and Baeza-Yates, 1992), cosine coefficient (Frakes and Baeza-Yates, 1992), distance-based measurements (Lee et al., 1989; Rada et al., 1989), and feature contrast model (Tversky, 1977). McGill etc. surveyed and compared 67 similarity measures used in information retrieval (McGill et al., 1979).

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 11–15,
Beijing, August 2010

3 Methodology

Similarity is a fundamental concept. Two documents can be said to be similar if both the documents have same content, describing a topic or an event or an entity. Similarity is a measure of degree of resemblance, or commonality between the documents.

In this work we have used Vector Space Model (VSM) for document representation. In VSM the documents are represented as vectors of unique terms. Here we have performed experiments by creating three types of document vector space models. In the first case we have taken all unique words in the document collection for vector of terms. In the second case we take the terms after removing all stop words. In the third case we have taken a sequence of words as terms. After the document model is built we use cosine similarity measure to identify the degree of similarity between documents.

In this work we have taken documents from the languages mentioned in the previous section. For the purpose of identifying similar documents across the languages we use map of term vectors of documents from English to other languages. Using the term vector map we can identify similar documents for various languages.

3.1 Similarity analyser

The main modules are i) Document vector creator ii) Translator and iii) Similarity identifier.

a) Document Vector Creator: Each document is represented as vector of terms. Here we take three types of term vectors. In the first type a single word is taken as a term which is the standard implementation of VSM. In the second type single words are taken but the stop words are removed.

In the third type each term is a sequence of words, where we define the number of words in the sequence as 4. This moving window of 4 is obtained by performing many experiments using different combinations of words. So our term of vector is defined as a set of four consecutive words, where the last three words in the preceding sequence is considered as the first three words in the following sequence. For example if a sentence has 10 words (w), the vector of terms for this sentence is w1w2w3w4, w2w3w4w5, w3w4w5w6, w4w5w6w7, w5w6w7w8, w6w7w8w9, w7w8w9w10. The weights of the terms in the vector are the term frequency and inverse document frequency (tf-idf). While creating document vectors, for Indian languages which are highly agglutinative and morphologically rich we use morphological analyzer to reduce the word into its root and it is used for document vector creation.

The first two experiments are the standard VSM implementation. The third experiment differs in the way the terms are taken for building the VSM. For building the VSM model which is common for all language document texts, it is essential that there should be translation/transliteration tool. First the terms are collected from individual language documents and a unique list is formed. The unique list of words is then translated using the translator module.

b) Word by Word Translator: In this module, the terms from English documents are taken and are translated to different languages. The translation is done word by word with the use of bilingual and multilingual synset dictionaries. This translation creates a map of terms from English to different languages. We have used bilingual dictionaries from English to Spanish, Hindi, Tamil, Telugu, and Malayalam dictionaries. Also we have used multilingual synset dictionaries for English, Tamil, Telugu, Hindi, and Malayalam. For each pair of bilingual dictionaries there are more than 100K root words. Since in this work we do not require syntactically and semantically correct translation of the sentences we adopted word to word translation. Hence we did not use any other system such as SMT for English to Indian languages. Named entities require transliteration. Here we have used a transliteration tool. This tool uses rule based approach, based on the phoneme match. The transliteration tool produces all possible transliteration outputs. Here we take into consideration the top five best possible outputs. For example the name “Lal Krishna Advani” would get transliterations in Indian languages as “laala krishna athvaani”, “laala krishna advaani”.

c) Similarity Identifier: The similarity identifier module takes the query in the form document as input and identifies all relevant documents. The similarity identifier uses cosine similarity measure over documents vector creator. The cosine similarity measure is the dot product of two vectors and is between 0 and 1 value. The more it is closer to 1, the similarity is more. The formula of cosine similarity is as follows:

$\mathrm{Sim}(S1,S2)_{tj} = \Sigma\ (W1j \text{ x } W2j\ )$ -- (1)

Where,

tj is a term present in both vectors S1and S2.
W1j is the weight of term tj in S1 and
W2j is the weight of term tj in S2.

The weight of term tj in the vector S1 is calculated by the formula given by equation (2), below.

Wij=(tf*log(N/df))/[sqrt(Si12+Si22+…+Sin2)] --(2)

Where,

tf = term frequency of term tj
N=total number of documents in the collection
df = number of documents in the collection that the term tj occurs in.
sqrt represents square root

The denominator [sqrt(Si12+Si22+……+Sin2)] is the cosine normalization factor. This cosine normalization factor is the Euclidean length of the vector Si, where 'i' is the document number in the collection and Sin2 is the square of the product of (tf*log(N/df)) for term in the vector Si.

4 Experiments and Results

We have performed three experiments with two different data sets. The first data set was collected by crawling the web for a single day's news articles and obtained 1000 documents from various online news magazines in various languages. The test set was taken from Times of India, The Hindu for English, BBC, Dinamani, Dinamalar for Tamil, Yahoo for Telugu, Matrubhumi for Malayalam, BBC and Dainik Jagran for Hindi and BBC for Spanish. The distribution of documents in the first set for various languages is as follows: 300 English, 200 Tamil, 150 Telugu, 125 Hindi, 125 Malayalam, 50 Spanish. The figure 1 given below shows the language distribution in this first set. The number of similar documents were 600 in this set.

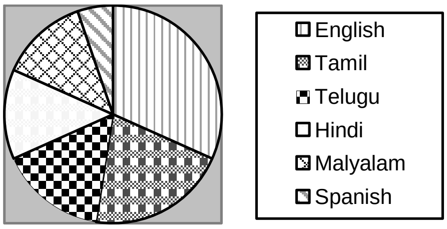

Figure 1. Data Distribution of Set 1

In the second data set we have taken news documents of one week time duration. This consisted of 9750 documents. The language distribution for this data set is shown in figure 2. This second data set consisted of 5350 similar documents.

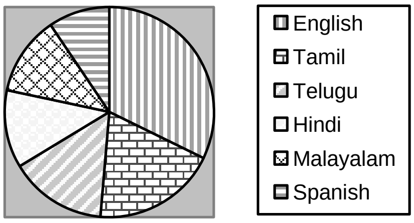

Figure 2. Data Distribution of Set 2

In the first experiment we took all the unique words (separated by white space) as terms for building the document vector. In the second experiment the terms taken were same as the first experiment, except that all the stop words were removed. In the third experiment, the terms taken for document vector creation were four consecutive words. The results obtained for three experiments for data set 1 is shown in Table 1. And results for data set 2 are shown in Table 2. Table 3 shows the similarity identification for various languages.

Here we take a news story document as a query and perform similarity analysis across all documents in the document collection to identify similarly occurring news stories. In the first data set in the gold standard there are 600 similar pairs of documents. And in the second data set there are 5350 similar pairs of documents in the gold standard.

It is observed that even though there were more similar documents which could have been identified, but the system could not identify those documents. The cosine measure for those unidentified documents was found to be lower than 0.8. We have taken 0.8 as the threshold for documents to be considered similar. In the documents which were not identified by the system, the content described consisted of less number of words. These were mostly two paragraph documents; hence the similarity score obtained was less than the threshold. In experiment three, we find that the number of false positives is decreased and also the number of documents identified similar is increased. This is because, in this case the system sees for terms of four words and hence single word matches are reduced. This reduces false positives. The other advantage of this is the words get the context, in a sense that the words in each sequence are not independent. The words get an order and are sensitive to that order. This solves sense disambiguation. Hence we find that it is solving the polysemy problem to some extent. The system can be further improved by creating robust map files between terms in different languages. The bilingual dictionaries also need to be improved.

In our work, since we are using a sequence of words as terms for document vectors, we do not require proper, sophisticated translation systems. A word by word translation would suffice to get the desired results.

Exp No	Gold std Similarity	System Identified Correct	System Identified Wrong	Prec %	Rec %
1	600	534	50	91.4	89.0
2	600	547	44	92.5	91.2
3	600	565	10	98.3	94.2

Table 1. Similarity Results on Data Set 1

Exp No	Gold Standard Similarity	System Identified Correct	System Identified Wrong	Prec %	Rec %
1	5350	4820	476	91.0	90.0
2	5350	4903	410	92.3	91.6
3	5350	5043	114	97.8	94.3

Table 2. Similarity Results on Data Set 2

Lang	Gold Std similar docs	System Identified correct	System Identified wrong	Prec %	Rec %
Eng	1461	1377	30	97.86	94.25
Span	732	690	15	97.87	94.26
Hin	588	554	11	98.05	94.22
Mal	892	839	19	97.78	94.05
Tam	932	880	22	97.56	94.42
Tel	745	703	17	97.63	94.36
AVG				**97.79**	**94.26**

Table 3.Similarity Results Data Set with Ex:3

5 Conclusion

Here we have shown how we can identify similar News document in various languages. The results obtained are encouraging; we obtain an average precision of 97.8% and recall of 94.3%. This work differs from previous works in two aspects: 1) no language preprocessing of the documents is required and 2) terms taken for VSM are a sequence of four words.

References

Frakes, W. B. and Baeza-Yates, R., editors 1992. *Information Retrieval, Data Structure and Algorithms*. Prentice Hall.

T. R. Gruber. 1993. *A translation approach to portable ontologies,* Knowledge Acquisition, 5(2):199–220.

Hindle, D. 1990. *Noun classification from predicate-argument structures*. In Proceedings of ACL-90, pages 268–275, Pittsburg, Pennsylvania.

Kohonen, Teuvo Kaski, Samuel Lagus, Krista Salojarvi, Jarkko Honkela, Jukka Paatero,Vesa Saarela, Anti. 2000. *Self organisation of a massive document collection*, IEEE Transactions on Neural Networks, 11(3): 574-585.

Lee, J. H., Kim, M. H., and Lee, Y. J. 1989. *Information retrieval based on conceptual distance in is-a hierarchies.* Journal of Documentation, 49(2):188–207.

McGill et al., M. 1979. *An evaluation of factors affecting document ranking by information retrieval systems.* Project report, Syracuse University School of Information Studies.

Rauber, Andreas Merkl, Dieter. 1999. *The SOMLib digital library system,* In the Proceedings of the 3rd European Conference on Research and Advanced Technology for Digital Libraries (ECDL'99), Paris, France. Berlin: 323-341.

Rada, R., Mili, H., Bicknell, E., and Blettner, M. 1989. *Development and application of a metric on semantic nets*. IEEE Transaction on Systems, Man, and Cybernetics, 19(1):17–30.

P. Resnik. 1995. *Using information content to evaluate semantic similarity in taxonomy,* Proceedings of IJCAI: 448–453.

Salton, Gerald. 1989. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer,* Reading, MA: Addison Wesley

Tversky, A. 1977. *Features of similarity*. Pychological Review, 84:327–352.