International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 11, November - 2013

Semantic Similarity Measure Using combination of Random Indexing and Latent Semantic Analysis.

Meenakshi P. Patil [1], S. K. Shirgave [2]

D. Y. Patil College of Engineering. & Technology.,Kolhapur,
Shivaji University, Kolhapur, Maharashtra, India

D. K. T. E"s Textile and Engineering Institute,
Rajwada, Ichalkaranji, Maharashtra, India

Abstract*: We present a method for measuring the semantic similarity of texts using a corpus-based measure of semantic similarity. This paper describes the use of different methods for semantic similarity calculation for predicting a specific type of textual coherence. We show that Random Indexing can be used to locate documents in a semantic space as well as terms, but not by straightforwardly summing term vectors. Using mathematical translation of the semantic space, we are able to use Random Indexing to assess textual coherence as well as LSA, but with considerably lower computational overhead. In this paper, we have combined two methods that are latent Semantic Analysis (LSA) and Random Indexing (RI) to increase the semantic similarity score to get greater extent*.

Keywords: Semantic Similarity, Random Indexing, Latent Semantic Analysis**.**

1. Introduction

Semantic similarity measures play important roles in information retrieval and natural Language Processing (NLP). Different methods are used for semantic similarity measure for predicting a specific type of textual coherence. The methods which are used in semantic similarity measure are:

- Random Indexing,
- Latent Semantic Analysis,

Also Natural Language Processing is used effectively for the same. Random Indexing is a vector based semantic representation model. It also uses Latent Semantic Analysis, a tool which is used to represent the meaning of words as vectors in a high-dimensional space. Text coherence is difficult in natural language processing.Coherence is a property of well-written texts that makes them easier to read and understand than a sequence of randomly strung sentences.

Random Indexing:

Generally Vector Space Model is used to represent semantic information about words, documents and other linguistic units. The idea is to use co-occurrence information to construct a multi-dimensional semantic space in which the linguistic units are represented by vectors whose relative distances represent semantic similarity between the linguistic units. The space is constructed by collecting co-occurrence information in a words-by-contexts frequency matrix where each row represents a unique word and each column represents a context (a word or a document). The frequency of co-occurrence with the given context is record by the cells of the co-occurrence matrix. As an alternative to vector-space models that use local co-occurrence matrices and some form of dimension reduction, the use of distributed representations that eliminates the need for separate dimension reduction of co-occurnce matrix. The technique, which is called Random Indexing, accumulates a words-by-contexts co-occurrence matrix by incrementally adding together distributed representations in the form of high-dimensional (i.e. on the order of thousands) sparse random index vectors. The index vectors contain a small number of non-zero elements, which are either +1 or -1, with equal amounts of both. For example, if the index vectors have eight non-zero elements, say, 1,800 dimensions, they have four +1s and four -1s. Depending on which kind of co-occurrences to use, the index vectors serve as indices or labels for words or documents. When using document-based co-occurrences, the documents are represented by high-dimensional sparse random index vectors, which are used to accumulate a words-by-contexts matrix by the following procedure: every time a given word occurs in a document, the document's index vector is added to the row for the word in the matrix. The procedure is similar when using word-based co-occurrences. First, we assign a high-dimensional sparse random index vector to each word type in the data. Then, every time a given word occurs in the data, the index vectors of the surrounding words are added to the row for the focus word. Words are thus represented in the co-occurrence matrix by high-dimensional context vectors that contain traces of every context (word or document) that the word has co-occurred with (or in).

Latent Semantic Analysis:

Latent Semantic Analysis (LSA) applies singular value decomposition (SVD) to the matrix. This is a form of factor

analysis, or more properly the mathematical generalization of which factor analysis is a special case. In SVD, a rectangular matrix is decomposed into the product of three other matrices. One component matrix describes the original row entities as vectors of derived orthogonal factor values, another describes the original column entities in the same way, and the third is a diagonal matrix containing scaling values such that when the three components are matrix-multiplied, the original matrix is reconstructed. Landauer, T. K., Foltz, P. W., & Laham, D.[8] make the statement that, there is a Mathematical proof that any matrix can be so decomposed perfectly, using no more factors. than the smallest dimension of the original matrix. When fewer than the necessary number of factors are used, the reconstructed matrix is a least-squares best fit. One can reduce the dimensionality of the solution simply by deleting coefficients in the diagonal matrix, ordinarily starting with the smallest. Generally a set of documents is conceptualized as a two-dimensional co-occurrence matrix, where the columns represent the documents and the rows represent the unique terms (usually words or short phrases) occurring in the documents. Sometimes every term appearing in the source document will be represented by a row, though it is more common to exclude a stop list of prepositions, function words, and other lexemes with negligible semantic content. The value in a particular cell may be a simple binary 1 or 0 (indicating the presence or absence of the term in the document) or a natural number indicating the frequency with which the term occurs in the document. Typically, each cell value is adjusted with an information-theoretic transformation. Such transformations widely used in IR weight terms. So that they more properly reflect their importance within the document. For example, one popular measure known as tf-idf (term frequency–inverse document frequency) uses the following formula:

$$W_{ij} = tf_{ij} log_2 \frac{N}{n_i}$$

Where,

W_{ij} is the weight of term i in document j,

tf_{ij} is the frequency of term i in document j,

N is the total number of documents, and

n_i is the number of documents in which i occurs.

After the weighting, pairs of documents can be compared by their column vectors, using some mathematical measure of vector similarity. Perhaps the most popular measure is the cosine coefficient,

$$cos(A, B) = \frac{\sum i A_i B_i}{|A||B|}$$

Some automatic summarization systems use the vector-space model to compare the semantic similarity of discourse units within a single document. In this case, the "documents" of the term–document co-occurrence matrix are actually sentences or paragraphs.

2. Literature Survey:

To quantify the concept of semantic similarity, some ideas have been put forth by researchers, most of which rely heavily on the knowledge available in lexical knowledge bases. Derrick Higgins and Jill Burstein [1] describe different methods for semantic similarity calculation for predicting a specific type of textual coherence. In this vector based semantic similarity measure is applied to the task of assessing a specific kind of textual coherence in student essay. Random Indexing (RI) or Latent Semantic Analysis (LSA) is used for the same task. Also predicted that if RI and LSA combined may yield superior results than the use of individual method. Ergin Altintas, Elif KArsligil and Vedat Coskun[2] introduced a new conceptual hierarchy based semantic similarity measure and evaluated it in word sense disambiguation (WSD) using a algorithm called Maximum Relatedness Disambiguation. It is used to increase the success rates of NLP applications. A concept hierarchy is used, so there is no sparse data problem in this approach. The main idea behind this evaluation is the success rate of WSD should increase as the similarity measure's performance gets better. Aminul Islam and Diana Inkpen [3] implemented a method for measuring the semantic similarity of texts using a corpus based measure of semantic word similarity and normalized and modified version of the Longest Common Subsequence (LCS) string matching algorithm. This method determines the similarity of two texts from semantic and syntactic information (in terms of common-word order) that they contain. In this first, string similarity and semantic word similarity are calculated and then uses an optional common-word order similarity function to incorporate syntactic information in this method, if required. Finally, the text similarity is derived by combining string similarity, semantic similarity and common-word order similarity with normalization. This proposed method is called the Semantic Text Similarity (STS) method. Benoit Lemaire and Philippe Dessus" [4] describes APEX (for an Assistant for Preparing Exams) which is a tool for evaluating student essays based on their content. It relies on a semantic text analysis method called Latent Semantic Analysis. LSA represent the meaning of the words as a vector in a high dimensional space. By comparing an essay and the text of a given course on a semantic basis, this system can measure how well the essay matches the texts. Various assessments are presented to the student regarding the topic, the outline and the coherence of the essay. Sanda M. Harabagiu and Dan I. Moldovan [9] describe a computational method that provides an explanation why a text is coherent .The computational method used is based on a parallel marker-propagation algorithm that is independent of the size of the knowledge base. The algorithm identifies paths in the knowledge base between the concepts of one clause and the concepts of the following clause. Actually some paths are eliminated when they don't satisfy the syntactic and semantic constraints of

the text. N. D. Karande and G. A. Patil [5] describe Natural Language Database Interfaces (NLDBI) which allows the user to query the database in a natural language. Architecture of new NLDBI system and its implementation with the result obtained is described. A natural language query is translated to an equivalent SQL query after processing through various stages.

3. Proposed Work

The main idea behind our enhancement in this approach is to achieve better semantic similarity measure. The basic idea in RI is to accumulate context vectors based on the occurrence of words in contexts (i. e. documents). LSA uses SVD, which is a matrix factorization technique that can be used to decompose and approximate a matrix so that the resulting matrix is much denser.

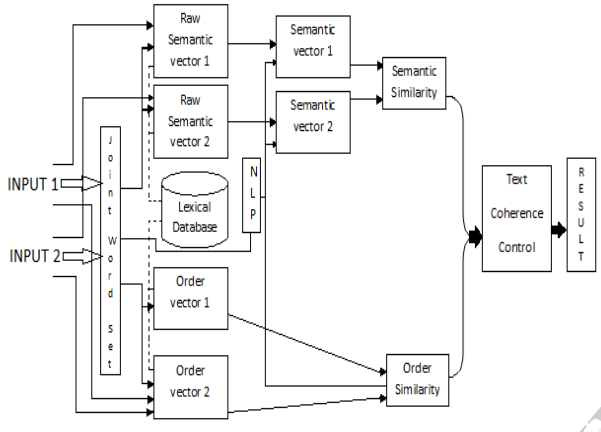

Figure1: Block Schematic for Proposed work

When Input 1 is given, the type of this input1 is specified with Input 2. Random indexing method is applied to the raw semantic vector block and from this semantic vectors are selected.

4. Methodology / Approach:

The straightforward method of applying Random Indexing to sentence similarity calculation may yield a maximum accuracy of 67.12%. To increase the accuracy of similarity some improvement has to be done over the exits method of random indexing. Suppose a set of random normalized term vectors are taken and produce a document vector to represent them. Then by summing the vectors and dividing by the number of vectors in the set, say n. As n increases, the document vector approaches the mean vector $x_{mean,}$ which is the average of all term vectors [1].

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \sim X_i = \sim X_{mean} \qquad \ldots\ldots\ldots\ldots\ldots (1)$$

This means that if we compare the similarity between two such random documents, as each document grows longer, the similarity should approach 1, since

$$\frac{\sim X_{mean} \cdot \sim X_{mean}}{\|\sim X_{mean}\|^2} = 1 \qquad \ldots\ldots\ldots\ldots\ldots (2)$$

Since the similarity between documents is bound to increase with their length, regardless of their relatedness, this may be a major problem. However, if the mean vector is subtracted from each of the term vectors, the bias from the system can be removed.

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} (\sim X_i - \sim X_{mean}) = 0 \qquad \ldots\ldots\ldots\ldots (3)$$

And 0.0 = 0

Subtracting the mean vector has the effect of reducing the magnitude of those term vectors which are close in direction to the mean vector, and increasing the magnitude of term vectors which are most nearly opposite in direction from mean vector. This means that , a document vector as a sum of term vectors will be created and those terms whose distribution is most distinctive will be given most weight, while terms which are less picky about what other terms they co-occur with will be given relatively little weight. This may achieves the effect of the inverse document frequency. This improved random indexing model may achieve the maximum accuracy of 70.1%. LSA method is used to select more semantic similar vectors. To implement LSA semantic space from a subset of the Wikipedia open-source encyclopedia, the SVDPACKC software package (Berryet al., 1993) [1], [7] may be used. The quality of the articles in this resource is somewhat variable. So to increase the quality of the training data, only encyclopedia articles with a single-word title (such as “Geography”) will be used. By using such provision, the computation which is required for large number of articles may get excluded. A tf-idf weighting scheme with log weighting of term frequencies and document frequencies are used to construct the document-by term matrix. Using this sentence similarity metric to predict sentences relatedness, the LSA model may achieves a maximum classification accuracy of 70.6%. Finally these two outputs are compared till its threshold value, and result will be generated more accurately. Thus, to get more semantic similar output the combination of RI & LSA is used. NLP will be used in case, the time factor is considered. To reduce the computational time, Semantic similarity will be calculated by using NLP. A natural language query is translated to an equivalent SQL query after processing through various stages. To process a query, speech tagging is the first step which is followed by the word tagging. Next step is the parsing the tagged sentence by a grammar. This grammar parser analyzes the query sentence according to the tag of each word and generates grammar tree. Finally the SQL translator processes the

grammar tree to obtain the SQL query. Probable grammar tree will be constructed which analyses the non terminals to collect the parameters which will be used in SQL. The SQL translator generates query in SQL. Using grammar, the parse tree is obtained from the input statement. The leaves of the parse tree are translated to corresponding SQL. The entire process involves tagging of input statement, apply grammar and semantic representation to generate parse tree, analyze the parse tree using grammar and translating the leaves of the tree to generate corresponding SQL query. The proposed method can be exploited in a variety of applications involving textual knowledge representation and knowledge discovery.

5. Results & Discussion:

For our experiments, we have collected different data sets from internet. We have carried out some experiments based on following methods a) Random indexing b) Latent Semantic Analysis c) combinations of both methods. We have obtained some useful results through these experiments. The results are: On the training data set (2236) and retrieved 1229 documents. Among these retrieved (1229). 894 are the most relevant document present in the data set.

Here following procedure is used to measure the performance of semantic similarity

• A data set contains **2236 documents**
• A search was conducted on that documents and **1229 documents** were retrieved.
• Of the **1229** documents retrieved, 894 were relevant.
Calculate the **precision** and **recall** scores for the search.

Using the designations above: As per the above discussion

A = The number of relevant documents retrieved,
B = The number of relevant documents not retrieved, and
C = The number of irrelevant documents retrieved.

Relevant document=A=894
B = Total documents - Relevant documents
= (2236 - 894) =1342
C = Retrieve documents - Relevant documents
= (1229 - 894) = 335

$$PrecisionRate(P) = \frac{A}{A+C} * 100$$

$$RecallRate(R) = \frac{A}{A+B} * 100$$

$$F_{measure} = \frac{P+R}{2}$$

Table 1: Performance of retrieved & relevant documents

Total Number of documents	2236
Total number of retrieved documents	1229
Total number of relevant documents	894
Precision rate	72.74
Recall rate	39.98
Fmeasure	56.36

6. Conclusion

The proposed system is concerned with implementing an intelligent system which will measure semantic similarity more accurately. The straightforward method of applying Random Indexing to sentence similarity calculation yielded a maximum accuracy of 70.1%, whereas LSA achieves a maximum accuracy of 70.6%. But using this new measure of sentence similarity i.e. the sum of the LSA and Random Indexing scores, maximum accuracy of 72.74%.

7. Future Scope:

The further work includes the implementation of other concepts of NLP, for example, Hidden Markov Model. Hidden Markav Model can be used to achieve a maximum accuracy. Hidden Markov models are especially known for their application in temporal pattern recognition such as speech, handwriting, part-of-speech tagging, partial discharges. A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other.

References

[1] "Sentence similarity measures for essay coherence" by Derrick Higgins, Educational Testing Service, Jill Burstein, Educational Testing Service, 7th International Workshop on Computational Semantics (IWCS), held January 2007.
[2] "A New Semantic Similarity Measure Evaluated Word Sense Disambiguation" by Ergin Altintas, Elif KArsligil and Vedat Coskun.
[3] "Semantic Text Similarity Using Corpus-Based Word Similarity and String Similarity" by Aminul Islam and Diana Inkpen, ACM Transactions on Knowledge Discovery from Data, Vol. 2, No. 2, Article 10, Publication date: July 2008.
[4] "A System to Assess the Semantic Content of Student Essays" by Benoit Lemaire and Philippe Dessus" J. Educational Computing Research, Vol. 24(3) 305-320, 2001.
[5] "Natural Language Database Interface for Selection of Data Using Grammar and Parsing" by N. D. Karande and G. A. Patil, World Academy of Science, Engineering and Technology 35 2009.

[6] H.H. Crokell, "Specialization and International Competitiveness," in Managing the Multinational Subsidiary, H. Etemad and L. S, Sulude (eds.), Croom-Helm, London, 1986. (book chapter style)

[7] Berry, M., Do, T., Krishna, G. O. V., and Varadhan, S. (1993). SVDPACKC (version 1.0) user's guide. University of Tennessee ms.

[8] "Introduction to Latent Semantic Analysis.Discourse Processes", 25, 259-284. Landauer, T. K., Foltz, P. W., & Laham, D. (1998).

[9] "A Marker-Propagation Algorithm for Text Coherence" by Sanda M. Harabagiu and Dan I. Moldovan