Constructing Better Document Vectors Using Universal Networking Language (UNL)

Chirag Shah
Dept. of CSE
IIT Bombay, Powai
Mumbai 400076,
Maharashtra, India.
Email: *chirag@cse.iitb.ac.in*

Bhoopesh Choudhary
Dept. of CSE
IIT Bombay, Powai
Mumbai 400076,
Maharashtra, India.
Email: *bhoopesh@cse.iitb.ac.in*

Pushpak Bhattacharyya
Dept. of CSE
IIT Bombay, Powai
Mumbai 400076,
Maharashtra, India.
Email: *pb@cse.iitb.ac.in*

Abstract

Encoding a document in a vector is a very crucial step for any vector space model based IR (Information Retrieval) system. It is obvious that the better these vectors are constructed, the better the performance of any application built on top of it. In traditional document representation methods, a document is considered as a bag of words. The fact that the words may be semantically related- a crucial information for document representation- is not taken into account. The feature vector representing the document is constructed from the frequency count of document terms. In this paper we describe a new method for generating feature vectors, using the semantic relations between the words in a sentence. The semantic relations are captured by the Universal Networking Language (UNL) which is a recently proposed semantic representation for sentences. In order to show that the generated document vectors with this new method are better than the traditional methods, we use the concept of mutual information. We prove by experiments that the vectors generated by UNL method indeed provide more information about the documents. It is proved that this helps in improving precision-recall in an IR system built using them.

1 Introduction

In recent years we have witnessed an ever-increasing flood of textual information on the Web [Lawrence and Giles, 1999]. Powerful search engines have been developed to aid locating documents by category, content, or subject. Many techniques are used to make such information retrieval effective like clustering [Jain *et al.*, 1999] or classifying [Joachims, 1997] documents, and finding similarity among them. In all of these applications, a very essential component is to represent documents in a format suitable for processing them. Representing documents using vectors is the most accepted and used method in data mining and IR (Information Retrieval) systems [Salton *et al.*, 1975]. The performance of any IR application based on vector space model depends highly on how well the documents are represented in terms of vectors. Many term-based methods for generating document vectors are known in IR community like binary vectors, TF (Term Frequency), TFIDF (Term Frequency with Inverse Document Frequency) [Salton, 1989], WIDF (Weighted IDF) [Tokunaga and Iwayama, 1994] etc. These methods consider the document as a bag of words, and do not exploit the relations that may exist between the words. One of the shortcoming of these methods is due to *polysemy* or *homography* where a word has different meanings or meaning shades in different contexts (for example, the word *bank* in *He went to the bank to withdraw some money* and *The boat was beside the bank*). It has been shown [Gonzalo *et al.*, 1998] that if we index words with their WordNet [Miller *et al.*, 1993] synsets or senses then it improves the information retrieval performance. Another problem with frequency-based methods is that they do not consider the structure of the sentences, which may also cause some problems (this has been discussed in section 2). Our work tries to improve the document representation by using the semantic information of the sentences representing the document. It uses Universal Networking Language (UNL) representation of a document, which presents the information given in a document in the form of a semantic graph.

The rest of the paper is organized as the following. Section 2 outlines the traditional way of generating document vectors and their shortcomings. Section 3 provides the details about constructing document vectors using semantics. In this section we describe Universal Networking Language (UNL) as a tool for using semantics in document representation. Section 4 covers the method for finding the *goodness* of document vectors. Experiments and results are reported in section 5. The paper is concluded in section 6.

2 Traditional Methods of Document Representation

The basic method of representing a document is by considering it an element in a vector space. Each component of the vector is the frequency of occurrence of a word in the document. The size of the vector can be reduced by selecting a subset of most important words according to some criterion. It is, however, a difficult problem to find a suitable subset of words that still represents the essential characteristics of the documents. It is also important to remove the words which are not informative, hence most common words like *and, with, to* etc., which are also known as stop words [Yang and Wilbur, 1996], are removed from the text while creating the vector. In addition to term frequency (TF), its IDF (Inverse Document Frequency) [Tokunaga and Iwayama, 1994] is also used to score a term. IDF of term t is defined as

$$IDF(t) = log\left(\frac{N}{N_t}\right) \tag{1}$$

where N is the total number of documents in the set and N_t is the number of documents in which term t occurs. The intuition behind IDF is that terms which rarely occur over a collection of documents are valuable [Tokunaga and Iwayama, 1994]. In other words, the importance of each term is assumed to be inversely proportional to the number of documents that contains the term. Many times TF and IDF are used in conjunction, i.e., a term's TF and IDF values are multiplied in order to get its score to be represented in the document vector. This method is referred to as *TFIDF method.*

It is expected that the representation of documents should reflect the knowledge meant to be conveyed by the documents. The above methods for representation of documents do not consider the semantic relations of the words. This may cause problems in many cases. For example, if we consider the sentences *John eats the apple standing beside the tree* and *The apple tree stands beside John's house*, they have almost the same set of words but talk about entirely different things. On the other hand there may be some sentences which have the same meaning but have been constructed from different sets of words. This case arises when synonymous words are used in the sentences. For example, in the sentences *John is an intelligent boy* and *John is a brilliant lad* mean more or less the same thing. There are some methods like Latent Semantic Indexing [Deerwester *et al.*, 1990] and Word Category Map in WEBSOM [Kohonen *et al.*, 2000], which try to address this problem. However, these methods highly depend on the statistical nature of the documents and do not implicitly exploit the semantic information carried by the documents for their representation.

In this paper we describe a **new method for the creation of document vectors**. This approach uses the Universal Networking Language (UNL) representation of a document. The UNL represents the document in the form of a set of semantic graphs with Universal Words (explained in the next section) as nodes and the semantic relation between them as links. Instead of considering the documents as a bag of words we use the information given by the UNL graphs to construct the vector.

3 Use of Semantics for Document Representation

As discussed above, there is a need for good syntactic and semantic analysis of the text to generate more *meaningful* document vectors. Here we describe use of UNL as a tool for such analysis and how to represent the documents using the rich information that it provides. We first give a brief introduction to UNL as well as the machine for generating it from the given text document. The method for constructing document vectors using UNL is then described.

3.1 Universal Networking Language (UNL)

Universal Networking Language (UNL) [Uchida *et al.*, 2000] is a semantic representation of a document, which expresses the document in the form of a graph. Information written in a natural language may be enconverted to UNL and the UNL can be deconverted into a target natural language. The UNL representation defines a semantic net [Woods, 1993] like structure. The meaning is represented sentence by sentence in the form of a hyper graph having concepts as nodes and relations as directed arcs. Concepts are represented as character-strings called Universal Words (UWs). The knowledge within a document is represented in three dimensions:

1. *Word Knowledge* is expressed by Universal Words (UWs), which are language independent. These UWs are restricted using constructs, which describe the sense of the word in the current context. For example, `drink(icl>liquor)` signifies that in the current context *drink* is a noun, which is a type of liquor. Here, *icl* stands for inclusion. *icl* restriction forms an *is-a* kind of relationship that is defined for semantic nets.
2. *Conceptual Knowledge* is captured by relating different Universal Words using the standard set of UNL Relation Labels. For example, *Humans are an intelligent species* is described as:

```
mod(species(icl>group), intelligent(icl>quality))
aoj(species(icl>group), human(icl>animal))
```

All relations in UNL are binary. A binary relationship between a pair of UWs is defined by *rel(UW1, UW2).* Here, *aoj* means agent with an attribute and *mod* restricts the scope of the entity specified as the first Universal Word `(species(icl>group))` (i.e., a restricted kind of species, which is intelligent).

3. *Speaker's view, aspect, tense of a verb, number of a noun etc.* are captured by UNL attributes. For example, consider the sentence *Please come here*, the UNL representation for which is

   ```
   plc(come(icl>do).@present.@request, here(icl>relative place))
   ```

   Here, `@request` describes the speaker's intention when he says *please*, `@present` means the present tense.

3.2 Generation of UNL

The machine for generating UNL from text is known as the *EnConverter* in the UNL parlance. The EnConverter is a language independent analyzer which provides a framework for morphological, syntactic and semantic analysis synchronously. It analyzes sentences by accessing a knowledge rich lexicon and interpreting the Analysis Rules. The process of formulating the rules is in fact programming a sophisticated symbol-processing machine.

The EnConverter can be likened to a multi-head Turing Machine. Being a Turing Machine, it is equipped to handle phrase structured (type 0) grammars and consequently the natural languages. The EnConverter delineates a sentence into a tree- called the nodenet tree- whose traversal produces the UNL expressions for the sentence. During the analysis, whenever a UNL relation is produced between two nodes, one of these nodes is deleted from the tape and is added as a child of the other node to the tree. The machine has two types of heads- processing heads and context heads. There are two processing heads- called Analysis Windows- and only the nodes under these take part in the analysis tasks like the generation of a UNL attribute or a relation. A node consists of the language specific word, the Universal Word and the attributes appearing in the dictionary as well as in the UNL expressions. The context heads are located on either sides of the processing heads and are used for look ahead and look back. The machine has functions like shifting the windows right or left by one node, adding a node to the node-list (tape of the Turing Machine), deleting a node, exchange of nodes under processing heads, copying a node and changing the attributes of the nodes. The complete description of the structure and working of the EnConverter can be found in [UNL, 2000]. Using this machine, the English Analyzer, which converts English sentences into UNL expressions has been built.

The English Analyzer makes use of the English-UW dictionary and the rule base for English Analysis. This rule base is for morphological, syntactic and semantic analysis. At every step of the analysis, the rule base drives the EnConverter to perform tasks like completing the morphological analysis, combining two morphemes, and generating a UNL expression. For want of space, the details are not given here. The reader is referred to [Shah *et al.*, 2000].

3.3 Document Vector Construction Using UNL

In the UNL method, instead of using the words as components for the document vector we use the Universal Words (UWs) as the components of the vector. Since each UW is disambiguated, multiple words in the document get automatically differentiated, thereby producing correct frequency count [Choudhary and Bhattacharyya, 2002]. The weight of each node in the graph is determined by the number of links to the node. The basic assumption behind this approach of counting the links is that *the more the number of links to and from a Universal Word, the more is the importance of the word in the document.* The weights are also determined subject to the following rules. All the relations in the UNL are divided in four categories based on their significance. By default each link representing the relation carries weight 1 except the partial transferable categories. The categorization is based on heuristics and our observations as well as experience with UNL.

1. **Transferable relation:** Under this category the weight of the parent node is added to the child node. The relations included are: *agt, obj, aoj, cag, cob, pur, ptn, rsn.*
2. **Equal weight relations:** Under this category the weight of the child node is made equal to that of parent node. The relations included are: *and, or, cnt, scn, pof, pos, coo, seq.*
3. **Partial transferable relations:** Under this category, weight is not transferred from the parent to child, rather the relation link is given more weight. The link weight of each relation is made 2 instead of default weight 1. The relations included are: *ben, ins, met, opl, plc, plf, plt, to, via.*
4. **Nontransferable relations:** Under this category, weight of the parent is *not* transferred to its child and also the link weight is the default weight 1. The relations in this category are *bas, con, dur, fmt, frm, gol, man, nam, per, qua, src, tim, tmf, tmt, mod.*

For example, consider the two sentences given in figures 1 and 2 as given documents. The vectors corresponding to the graphs are:

$$X_1 =< 4, 3, 2, 3, 4, 0, 0 > \text{ and } X_2 =< 4, 0, 0, 0, 4, 3, 2 >$$

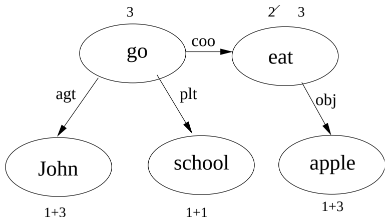

Figure 1: UNL graph of the sentence *John is going to the school eating an apple*

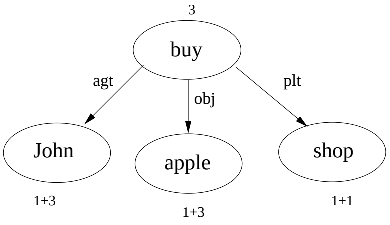

Figure 2: UNL graph of the sentence *John bought the apple from the shop*

Here, the words considered are *John, go, school, eat, apple, buy* and *shop* in that order. The number for each word in the vectors represents the weight calculated using the method described above and demonstrated in the figures.

In order to incorporate the importance of a UW based on the entire database as in done in TFIDF, we can also find IDF for UW here. The process of construction of the document vector from the UNL representation is summarized as the following.

1. Parse the UNL document to construct the UNL graph.
2. For each UW in the UNL graph count the links to/from other UWs from/to it.
3. Adjust the count depending upon the relation connecting given UWs.
4. Merge all the counts of each UW for a given document.
5. Multiply this count with its corresponding IDF.
6. Construct the document vector by assigning counts of UWs to their corresponding positions in the vector.

It is important to note that this proposed approach does not lose any information given by the word frequency method, since the method implicitly incorporates the frequency of the UWs. *For any node in the graph there is at least one link incident on it.*

4 Finding *Goodness* of Document Vectors

It is very essential to check how well the documents are represented in terms of vectors by a particular scheme. Typically, researchers build the whole IR system and find precision-recall parameters [Croft, 1995]. This kind of evaluation method requires human judgments about the relevance of the documents to the queries. It has some disadvantages like difficulty in getting enough amount of relevance judgments by humans [Jones and van Rijsbergen, 1975], and unreliability of such judgments [Mizzaro, 1999]. We, therefore, used the method proposed by Rong Jin et al [Jin *et al.*, 2001] to find the *goodness* of document vectors. This section describes the proposed method in brief.

4.1 Intuition Behind Using Mutual Information for *Goodness* Measurement

As shown by Shannon in his classical work on information theory [Shannon, 1948], entropy of an event C can be defined using its probability distribution p as

$$H(C) = -\sum_{i=1}^{l} p_i \cdot log(p_i) \tag{2}$$

where l is the total number of discrete states. The conditional entropy $H(C|D)$ can be calculated as

$$H(C|D) = H(C,D) - H(D) \tag{3}$$

This conditional entropy can be found as

$$H(C|D) = -\sum_{i=1}^{l}\sum_{j=1}^{l} p_i \cdot p_{(i|j)} \cdot log(p_{(i|j)}) \tag{4}$$

According to the definition [Press *et al.*, 1993], the mutual information *I(C,D)* can be represented as

$$\begin{aligned} I(C,D) &= H(C) + H(D) - H(C,D) && (5)\\ &= H(C) + H(D) - H(C|D) - H(D) \qquad \text{(from equation (3))} && (6)\\ &= H(C) - H(C|D) && (7) \end{aligned}$$

In our case, the two random variable, C and D, correspond to the *document content* and the document vectors respectively. Therefore, *H(C)* represents the uncertainty in guessing the content of a document given that we only know that the document is in the collection, while the conditional entropy *H(C|D)* measures the uncertainty about the document content given that we are allowed to look at the representation vector for the document. The difference between these two entropies, i.e., the mutual information *I(C,D)*, tells us how much more confidence that we gain in guessing the document content after looking through the document vectors. Thus, the mutual information *I(C,D)* reflects the *informativeness* of the document vectors generated by the term weighing schemes and gives the sense of *goodness* of the term weighing schemes.

4.2 Mathematical Description

Here we are providing the necessary mathematical formulation only. The reader is referred to [Jin *et al.*, 2001] for more details.

Let n be the number of documents in the collection. Let $d_1, d_2, ..., d_n$ be the document vectors in term space. Let M be the document-term matrix. Each number M_{ij} in the matrix M represents the weight of the j^{th} word in the i^{th} document. Let D be the document-document matrix and it is defined as

$$D = MM^T \tag{8}$$

Let C be the random variable for *document content*. This *document content* can be represented as a set of weighted *concepts* and each *concept* corresponds to an eigenvector of the document-document matrix D. Therefore, the random variable C is essentially related to and can be defined in the following way: the random variable C can only take one of the values from the set of eigenvectors $v_1, v_2, ..., v_n$ and the eigenvalue λ_i indicates the importance of the eigenvector v_i. Therefore, we can assume that the probability for the random variable C to be the eigenvector v_i is proportional to the eigenvalue λ_i.

$$P(C = v_i) = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j}, 1 \leq i \leq n \tag{9}$$

The random variable D represents the document vector. The possible values that it can take are the set of document vectors in the document collection, i.e., $d_1, d_2, ..., d_n$. Since every document in the collection is equiprobable, we can assume the uniform distribution for the random variable D, that is, the probability for the random variable D to be any document vector d_i is a constant, or

$$P(D = d_i) = \frac{1}{n}, 1 \leq i \leq n \tag{10}$$

Now, the document can be viewed as a set of *concepts* and the weight for each *concept* is given by the projection of the document vector on the corresponding axis. Therefore, we can assume that the probability for a document to contain some particular *concept* is proportional to the projection of the document vector on the corresponding *concept* axis. Thus, the conditional probability $P(C = v_i|D = d_j)$ would be proportional to the projection of document vector d_j on the *concept* axis v_i, that is:

$$P(C = v_i|D = d_j) = \frac{|d_j^T v_i|}{\sum_{k=1}^{n} |d_j^T v_k|} \tag{11}$$

With all these probabilities defined, we can find their respective entropies and finally, the mutual information as defined in equation (7). The more this mutual information for a given method of vector generation, the better that method.

5 Experiments and Results

We selected 82 documents from various sources as shown in Table 1 for our experiments. Since the fully automated system for converting documents in UNL was not available at the time of doing these experiments, we generated and verified UNL expressions for each document manually.

Class	**Number of documents**
Barcelona corpus	14
ITU corpus	8
Legal	8
Health	4
Politics	19
Economics	24
Technical manual	5
Total	82

Table 1: Corpora Used for Experiments

After generating document vectors using various methods (TF, TFIDF, UNL-UW, UNL-UW with IDF), we found their *goodness* using the method described in the previous section. The results are given in Table 2. Following observations can be made from these results:

Following observations can be made from these results:

Method	**H(C)**	**H(C|D)**	**I(C,D)**
TF	1.9863	5.8887e-4	1.9857
TFIDF	4.0780	5.3013e-4	4.0775
UNL (UW only)	4.0587	5.7771e-4	4.0581
UNL (UW with IDF)	4.1701	5.5390e-4	4.1696

Table 2: Mutual Information for Various Term Weighing Schemes

1. TFIDF performs better than TF.
2. UNL-based methods perform better than frequency-based methods.
3. In UNL-based methods also, considering IDF helps.

6 Conclusion

We have proposed a new method for document vector construction. This method uses the semantic information present in the form of relations between words in sentences. Thus the approach is different from traditional methods of document vector generation, which consider the document as a bag of words. As shown in the experiments, this approach performs better than tradition TF and TFIDF methods. This *goodness* is proved by using the concept of mutual information. It is shown in [Jin *et al.*, 2001] that this measure is highly correlated with the precision-recall measurements in a typical IR system. Therefore, even-though we have not applied the prepared document vectors for any particular IR application, we have shown that they provide better representation of the documents so as to help further in any application built on top of them.

References

[Choudhary and Bhattacharyya, 2002] Bhoopesh Choudhary and Pushpak Bhattacharyya. Text clustering using semantics. In *The Eleventh International World Wide Web Conference*, 2002.

[Croft, 1995] W. Bruce Croft. What do people want from information retrieval? *D-Lib Magazine*, November 1995.

[Deerwester *et al.*, 1990] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 1990.

[Gonzalo *et al.*, 1998] Julio Gonzalo, Felisa Verdejo, Irina Chugur, and Juan Cigarra. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP, Montreal.*, 1998.

[Jain *et al.*, 1999] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, 1999.

[Jin *et al.*, 2001] Rong Jin, Christos Falusos, and Alex G. Hauptmann. Meta-scoring: automatically evaluating term weighting schemes in IR without precision-recall. In *Proceedings of the 24th annual* *international ACM SIGIR conference on Research and development in information retrieval*, pages 83–89. ACM Press, 2001.

[Joachims, 1997] Thorsten Joachims. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 143–151, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.

[Jones and van Rijsbergen, 1975] K. Sparck Jones and C. van Rijsbergen. Report on the need for and provision of an "ideal" information retrieval test collection. Technical Report British Library Research and Development Report 5266, Computer Laboratory, University of Cambridge, 1975.

[Kohonen *et al.*, 2000] T. Kohonen, S. Kaski, K. Lagus, J. Salojrvi, V. Paatero, and A. Saarela. Self organization of a massive document collection. *IEEE Transactions on Neural Networks*, 11(3):574–585, May 2000. Special Issue on Neural Networks for Data Mining and Knowledge Discovery.

[Lawrence and Giles, 1999] Steve Lawrence and C. Lee Giles. Searching the Web: General and Scientific Information Access. *IEEE Communications Magazine*, January 1999.

[Miller *et al.*, 1993] George A. Miller, Richard Beckwith, Christian Fellbaum, Derek Gross, and Katherine Miller. *Introduction to WordNet: An On-line Lexical Database.* Cognitive Science Laboratory, Princeton University, August 1993.

[Mizzaro, 1999] S. Mizzaro. Measuring the agreement among relevance judges. In *MIRA99*, Glasgow, UK, 1999.

[Press *et al.*, 1993] W.H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1993.

[Salton *et al.*, 1975] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620, 1975.

[Salton, 1989] Gerald Salton, editor. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer.* Addison-Wesley, 1989.

[Shah *et al.*, 2000] Chirag Shah, Jignashu Parikh, and Trilok Soni. Conversion of English Language Texts to Universal Networking Language. *B.E. Dissertation, Dharamsinh Desai Institute of Technology, Nadiad*, 2000.

[Shannon, 1948] Claude Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, pages 379–423, 623–656, July, October, 1948.

[Tokunaga and Iwayama, 1994] T. Tokunaga and M. Iwayama. Text categorization based on weighted inverse document frequency. Technical Report 94 TR0001, Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan, 1994.

[Uchida *et al.*, 2000] H. Uchida, M. Zhu, and S. T. Della. UNL: A gift for a millennium. Technical report, The United Nations University, 2000.

[UNL, 2000] UNL Centre, UNU, Tokyo 150-8304, Japan. *EnConverter Specification Version 2.1*, 2000.

[Woods, 1993] William A. Woods. What's in a link: Foundation for semantic network. *Journal of Documentation*, 49:188–207, 1993.

[Yang and Wilbur, 1996] Yiming Yang and John Wilbur. Using corpus statistics to remove redundant words in text categorization. *Journal of the American Society of Information Science*, 47(5), 1996.