Improving Document Vectors Representation using Semantic Links and Attributes

Abstract

Document representation is a crucial step in any Information Retrieval (IR) system. Since most of the traditional methods do not consider much of semantic or syntactic information, the representation becomes insufficiently informative for an IR task. We describe a novel approach to incorporating Natural Language Processing (NLP) in document representation for addressing this problem. Use is made of additional information about the sentences, viz., (i) syntactic links among the words found by the Link Parser and (ii) heuristically determined semantic attributes of the words. After mapping this information to the document level using Self-Organizing Map (SOM), we use it for embellishing the document vectors constructed by the TFIDF method. The efficacy of the proposed method is established by showing that the document vectors (i) have higher mutual information content *and (ii) achieve better class separation.*

Keywords: Natural Language Processing (NLP), Information Retrieval (IR), text representation, Link Grammar, semantic attributes, Self-Organizing Map (SOM)

1 Introduction

The use of syntax and semantics for information retrieval (IR) is well studied in the literature. While many researchers accept the fact that in principle, such additional information should help in improving IR [17], there are divergent views on whether and how to use this information. For instance, Bowen Hui [7] examined some limitations of traditional IR systems and laid out the motivation for applying NLP techniques to IR, focusing at the same time on only morphological processing and stressing that *"not all natural language phenomena apply to IR"*. Now the question is what phenomena are important.

In this paper we investigate the effect of some of such phenomena like link information and word attributes on IR. It is obvious that the *better* the documents are represented in terms of vectors, the better will be the performance of an IR system built using them. We, therefore, focus on representing the documents in a *better* way using with some NLP techniques in this paper. We argue that *incorrectness* and *insufficiency* of the information are two major drawbacks of traditional document representation schemes and focus on addressing the problem of *insufficiency* of the information. The rest of the paper is mainly organized in four parts. Part I sets the motivation behind using NLP for IR in general and link and attributes information in particular. Part II describes our method for collecting additional information about the documents using Link Grammar [24] and other heuristics. This part demonstrates how relations and attributes information can help in supplying some additional and useful information about the documents. In order to measure the *goodness* of generated document vectors, we follow the intuition and compelling experimental evidences provided by Rong Jin *et al.* [8] that *the more informative the document vectors are, the better will be the performance of IR using these vectors.* We, therefore, find the *informativeness* of the document vectors, which is explained in part III of the paper. In part IV we provide additional support for our method by finding interclass distance for traditional method and our proposed method. In section 9 we conclude that careful use of NLP can indeed improve performance of an IR system.

I Motivation behind using NLP for IR

In this part we analyze the shortcomings of traditional methods of document representation and set the motivation for using NLP for IR in general, and document representation in particular. Here we are considering vector space model [20], which is the most accepted and widely used method for document representation.

2 Shortcomings of the Traditional Methods

It is expected that the representation of documents should reflect the knowledge meant to be conveyed by the documents. The traditional methods for representation of documents like Term Frequency (TF) [21], Term Frequency with Inverse Document Frequency (TFIDF) [9], Weighted IDF (WIDF) [25] *etc.* do not consider the senses of the words or their mutual semantic relations. This causes problems. For example, the two sentences *John is eating the apple, standing beside the tree* and *The apple tree stands beside John's house* have the same set of content words (except *house*), but mean entirely different things. On the other hand, the sentences *John is an intelligent boy* and *John is a brilliant lad* mean almost the same thing. In general, the vector space model of document representation [20] using *bag of words*, suffers from two problems:

1. **Incorrectness of information**
   Synonymy (more than one word having the same sense), and *polysemy* (single word having more than one sense) affect recall and precision respectively [4]. There are many studies for solving these problems using Word-Sense Disambiguation (WSD) [14, 1, 15]. However, [22] reports that disambiguation accuracy of at least 90% is required to avoid the degradation of effectiveness of retrieval.

2. **Insufficiency of information**
   Many times merely considering the words of a document may not be enough for representation as a document is not a bag of words. For example, consider two documents using the same set of words, but one talks in a positive sense, while the other talks in a negative sense. Since the traditional methods of document representation do not have the mechanism to capture the tone or the structure of the document, both of these documents will get almost the same vector representation. This results in coarse clustering or poor precision when such vectors are used in an IR system. It is obvious that a document is represented more appropriately if syntactic and semantic information is included.

   The present paper is concerned with this issue of insufficiency of information.

II Using Relations and Attributes for Extending Document Vectors

In this part we describe a method for using relations and attributes for capturing some semantic information about the documents that can be used to augment the TFIDF document vectors. Since Link Grammar considers a sentence as the unit, we first represent the sentences using Self-Organizing Map (SOM) according to their relations and attributes (section 4). This map, called *Sentence Category Map (SCM)*, is then used to construct the extended portion of the document vector for each document (section 5).

3 Motivation for using Semantic Relations and Attributes

Consider the following sentences:

1. *John plays football.*
2. *John did not play football.*
3. *John likes to play football.*
4. *John cannot play football.*
5. *He asked John to play football.*

Traditional frequency based methods represent all these sentences[1] in the same way. However, there are differences among these sentences with respect to the tense, intention or ability of the agent, positive or negative connotation, semantic roles of the words and so on. These details demand a deeper analysis of the text. To facilitate this, we use Link Grammar [24] as one of the tools, which provides the link information among the words of a sentence. For example, for the sentence *John plays football*, the Link parser outputs

```
  +--Ss-+----Os---+
  |     |         |
John plays.v football.n
```

where *John* is the subject of *play* and *football* the object. Additionally, we make use of heuristics that capture semantic attributes. Table 1 show the list of attributes along with their explanations.

[1] All words other than *John, play,* and *football* are eliminated as stop words [26].

Attribute	Meaning
not	Negative sense
present	Present tense
past	Past tense
future	Future tense
def	Definite
indef	Indefinite
contrast	Shows contrast in statements
ability	Demonstrates ability of some act
should	To do something as a matter of course
may	Possibility that something is true or happens
just	Expresses an event or a state that has just begun or ended
yet	Expresses an event or a state that has not yet begun or started
progress	An event is in progress
request	Request for something

Table 1: Semantic attributes and their meanings

4 Forming Sentence Category Map (SCM)

Since extracting links information and heuristic analysis is done at the sentence level, we need to extend it to the document level in order to obtain document representation. We make use of Self-Organizing Map (SOM) for this purpose. SOM falls under a special class of neural networks based on *competitive learning* [5]. In such neural networks, output neurons of the network compete among themselves to be activated or fired, with the result that only *one* output neuron, or one neuron per group, is on at any one time. An output neuron that wins the competition is called a *winner takes all neuron* or simply a *winning neuron* [27]. Various characteristics like approximation of the input space, topological ordering, density matching, and ability to select the best features, make SOM ideal for performing tasks like pattern organization [5].

In order to use SOM for organizing sentences, we partly follow the design of WEBSOM [6, 12] - a very famous implementation of SOM for document clustering. The first stage of WEBSOM consists of forming the *word category map* (WCM), which is also referred to as *semantic SOM* [19] or *contextual map* [11]. The input to this stage is a set of patterns. These patterns are prepared for words using their average short context. In our case, we need to encode the sentences to generate input patterns. This step along with the procedure for training the SOM (as given by Kohonen [11]) is given in the following algorithm.

1. *Input patterns generation:* There are 107 Link Grammar relations and 14 attributes that we generate. Therefore, form a vector of size 121 for each sentence assigning. Each component of this vector will denote the count for the corresponding link or attribute in that sentence.

2. *Initialization.* Seclect the size of the SOM. We chose 20x20 (400 neurons). With each neuron j, there will be a weight vector w_j associated. Choose random values for the initial weight vectors $w_j(0)$. The only restriction here is that the $w_j(0)$ be different for $j = 1, 2, ..., l$, where l is the number of neurons in the lattice. It may be desirable to keep the magnitude of the weights small.

3. *Sampling.* Draw a sample X from the input space of N vectors.

4. *Similarity Matching.* Find the best-matching (winning) neuron $i(X)$ at time step n by using the Euclidean minimum distance criterion

$$i(X) = arg\left(\min_j ||X(n) - w_j(n)||\right), j = 1, 2, ..., l \quad (1)$$

5. *Updating.* Adjust the weight vectors of all neurons by using the update formula

$$w_j(n+1) = w_j(n) + \eta(n)h_{j,i(X)}(n)(X(n) - w_j(n)) \quad (2)$$

where $\eta(n)$ is the learning-rate parameter, and $h_{j,i(X)}(n)$ is the neighborhood function (we used *Gaussian*) centered around the winning neuron $i(X)$; both $\eta(n)$ and $h_{j,i(X)}(n)$ are varied dynamically during learning for the best results as shown in [5].

6. *Continuation.* Continue with steps 3, 4, and 5 until no noticeable changes in the feature map are observed.

This scheme is shown in Figure 1. We chose to train a SOM of size 20 × 20. The results of various stages of the algorithm are shown in Figures 2, 3, and 4. These figures represent the effect of input patterns on the SOM. Since initially the weight vectors corresponding to neurons are initialized to some random values, the input patterns get mapped at any position on the map. As the ordering of weight vectors according to the input patterns takes place, we can see a

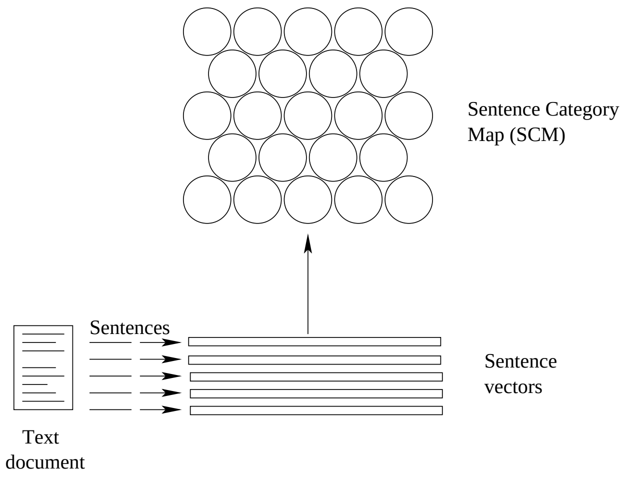

Figure 1: SCM formation

gradual organization in the map. More details about this procedure can be found in [11, 5].

After the map is constructed, we can observe a kind of organization of sentences according to their semantic representation given by the relations and the attributes. Therefore, we call it *Sentence Category Map (SCM).*

5 TFIDF Vectors Enhancement with SCM

Once the sentences are organized using SOM, we can find the representation for the documents using the following algorithm.

1. For every document do the following. Take every sentence's vector and input it to the trained SCM. Find the winning neuron. This constitutes one *hit* on that neuron.

2. Collect hit information from all the neurons and construct a vector using it. In our case this vector is of size 400.

3. Append this vector to the normal TFIDF vector for the document.

This scheme is shown in Figure 5.
The resulting vectors incorporate *additional* and *useful* information about the document. They are expected to give *better semantic representation* of the document. This *goodness* is justified in the next two parts of the paper.

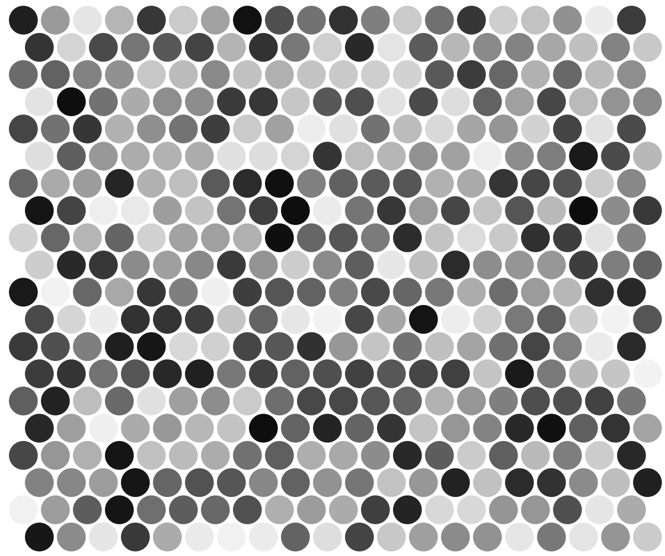

Figure 2: Initialization

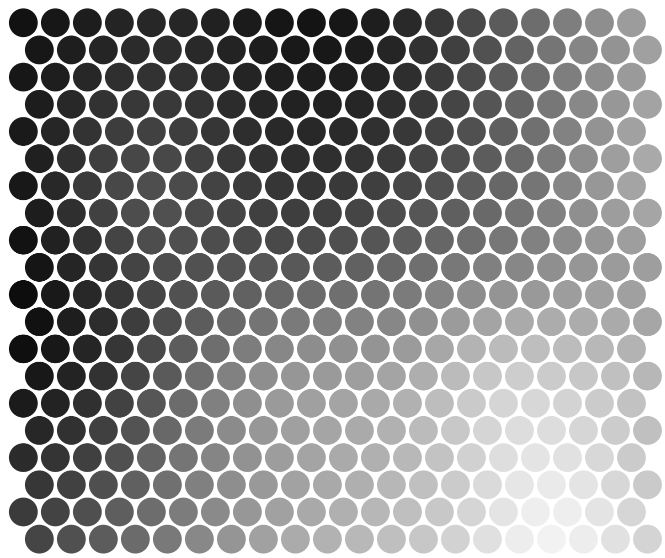

Figure 3: Ordering

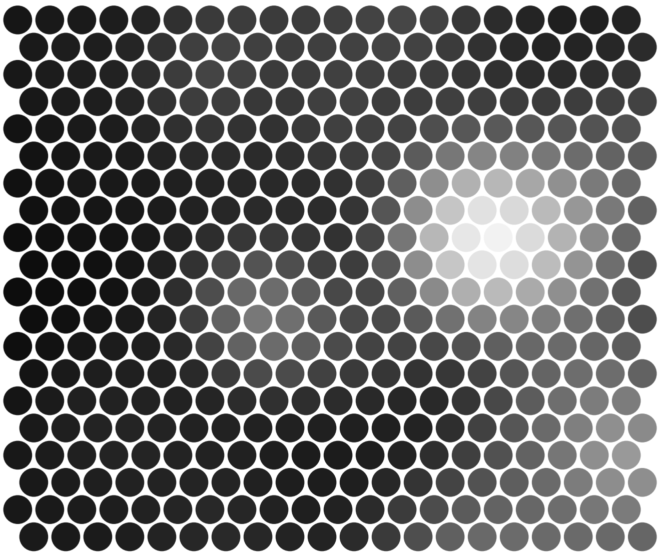

Figure 4: Fine tuning

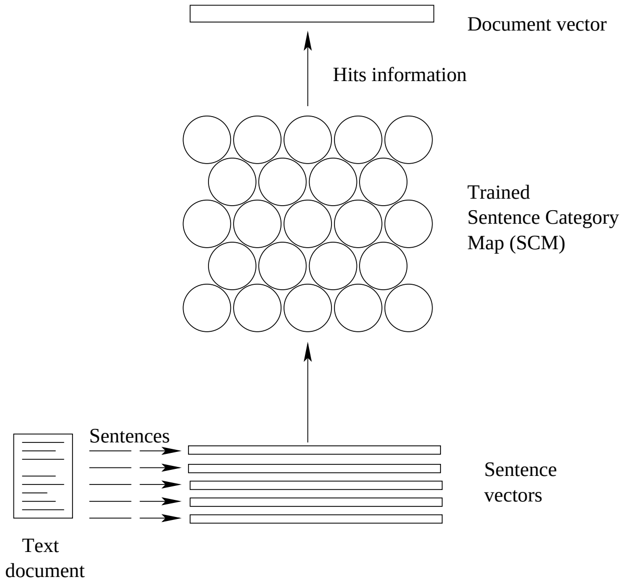

Figure 5: Extending document vectors

III Evaluating the Goodness of Document Vectors using Information Content

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 [2]. 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 [10], and unreliability of such judgments [16]. We, therefore, used the method proposed by Rong Jin *et al.* in their SIGIR 2001 paper [8] to find the *goodness* of document vectors. The method is derived from the concepts of Latent Semantic Indexing (LSI) [3] and information theory [23]. This part describes the proposed method in brief with our experiments and results.

6 Intuition Behind Using Mutual Information for Goodness Measurement

In this section we analyze how to measure the information content of document vectors for evaluating their *goodness*. In order to understand this, we need to enumerate few concepts from information theory. As shown by Shannon [23] in his classical work on information theory, entropy of an event C can be defined using its probability distribution p_i as

$$H(C) = -\sum_{i=1}^{l} p_i \cdot log(p_i) \tag{3}$$

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

$$H(C|D) = H(C,D) - H(D) \tag{4}$$

where $H(C,D)$ is the joint entropy of events C and D. In particular, the conditional entropy can be found using

$$H(C|D) = -\sum_{i=1}^{l}\sum_{j=1}^{l} p_j \cdot p_{(i|j)} \cdot log(p_{(i|j)}) \tag{5}$$

where p_j is the probability distribution of random variable D and $p_{(i|j)}$ is the conditional probability distribution for random variable C given random variable D.

Now we describe how to measure information content using entropy. According to the definition [18], the mutual information *I(C,D)* can be evaluated as[2]

[2]More information about how the information content is related to the entropy can be found in [23, 13].

$$
\begin{aligned}
I(C,D) &= H(C)+H(D)-H(C,D) && (6)\\
&= H(C)+H(D)-H(C|D)-H(D) && (7)\\
&\quad \text{(from equation 4)} \\
&= H(C)-H(C|D) && (8)
\end{aligned}
$$

i.e., the difference between $H(C)$, the entropy of the random variable C, and $H(C|D)$, the average entropy of the random variable C given the value of the random variable D. The entropy of a random variable C represents the uncertainty in guessing the value of the random variable C. Therefore, the mutual information *I(C,D)* measures the *decrease of the uncertainty* in the value of the random variable C caused by knowing the value of the random variable D.

Linking this understanding to the document representation case, the two random variables C and D correspond to the *document content* (described in the next section), 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 provided the representation vectors for the documents. The difference between these two entropies, *i.e.*, the mutual information *I(C,D)*, indicates the decrease of the uncertainty about knowing the document content. In other words, it tells us how much more confidence we gain in guessing the document content after seeing the document vectors. Thus, the mutual information *I(C,D)* reflects the *informativeness* of the document vectors generated by the term weighing schemes giving the sense of *goodness* of these schemes.

The intuition that more informative vectors help in improving IR is verified by Rong Jin *et al.* [8] through large scale experiments. The authors conducted their experiments on four different term weighing schemes over six different collections. These test collections were taken from TREC, the size of which varied from small collection with 20,000 documents to fairly large collection with 160,000 documents. The average size of the document also varied very much from collection to collection. In such diversity of the corpora also they found that the average precision measures were quite consistent with mutual information in every single case. These compelling experiments prove that *the more informative the document vectors are, the better will be the performance of IR using these vectors.*

7 Mutual Information of Document Vectors

In this section we formalize the ideas expressed in the previous section. Here we are providing the necessary mathematical formulation only. The reader is referred to [8] 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, which can be found as

$$D = MM^T \tag{9}$$

As defined earlier, C is the random variable for *document content*. We define *document content* as a set of weighted *concepts* and each *concept* corresponds to an eigenvector of the document-document matrix D. Thus, 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, which enables us to define the probability distribution for random variable C as the following.

$$P(C = v_i) = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j}, 1 \le i \le n \tag{10}$$

The random variable D corresponds to the document vectors. 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

$$P(D = d_i) = \frac{1}{n}, 1 \le i \le n \tag{11}$$

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:

$$P(C = v_i | D = d_j) = \frac{|d_j^T v_i|}{\sum_{k=1}^{n} |d_j^T v_k|} \tag{12}$$

With all these probabilities defined, we can find their respective entropies and finally, the mutual information as defined in equation 8. The more this mutual information for a given method of vector generation, the better is that method.

8 Goodness using Mutual Information

We used British National Corpus (BNC) for our experiments. It is a 100 million word collection of samples from a wide range of sources, designed to represent a wide cross-section of current British English. The text corpus includes extracts from regional and national newspapers, specialist periodicals and journals for all ages and interests, academic books and popular fiction, published and unpublished letters and memorandums, school and university essays, among many other kinds of text. The detail of the documents used for our experiments is given in the following table.

Class	Number of docs
Applied science	60
Arts	121
Belief	69
Commerce	92
Imaginative	114
Leisure	180
Natural science	49
Social science	202
World affairs	213
Total	1,100

Table 2: BNC Documents used

Method	$H(C)$	$H(C\|D)$	$I(C,D)$
TFIDF	5.5818	2.8458e-6	5.5818
Extended TFIDF	5.6664	2.4029e-6	5.6664

Table 3: Mutual information for various term weighing schemes

The following observations can be made from the above results:

1. The $H(C|D)$ value of extended TFIDF method is less than normal TFIDF method, which means that the uncertainty of guessing the document content with our method is less than the traditional TFIDF method.

2. Our method gives more mutual information showing *better* representation of the documents.

IV Evaluation using Interclass Distance

As shown in table 2 we used 1,100 documents from nine classes of BNC. After obtaining the vector representation of these documents using TFIDF and our proposed method described in this paper, we found interclass distances for them. The results for both these schemes are given in tables 4 and 5. They are also summarized in figure 6. As can be seen from the results, our method is able to separate classes better than the traditional TFIDF method.

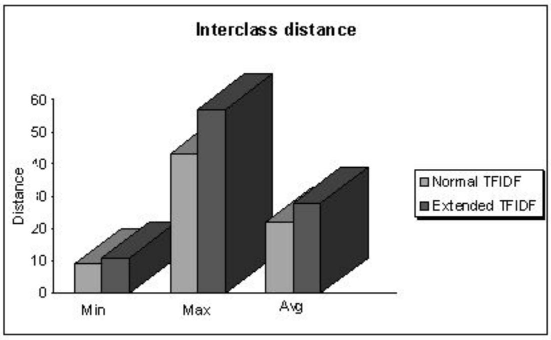

Figure 6: Interclass distance

Class	1	2	3	4	5	6	7	8	9	Min	Max	Avg
1	0.00	28.25	8.63	20.04	26.59	36.75	13.80	39.51	41.55	8.63	41.55	23.90
2	28.25	0.00	21.44	9.93	3.35	10.61	39.83	16.98	21.01	3.35	39.83	16.82
3	8.63	21.44	0.00	13.16	19.79	30.00	20.45	33.21	35.56	8.63	35.56	20.25
4	20.04	9.93	13.16	0.00	8.20	18.69	31.74	23.12	26.32	8.20	31.74	16.80
5	26.59	3.35	19.79	8.20	0.00	12.18	38.21	18.07	21.92	3.35	38.21	16.48
6	36.75	10.61	30.00	18.69	12.18	0.00	48.29	12.95	17.79	10.61	48.29	20.81
7	13.80	39.83	20.45	31.74	38.21	48.29	0.00	50.52	52.19	13.80	52.19	32.78
8	39.51	16.98	33.21	23.12	18.07	12.95	50.52	0.00	12.49	12.49	50.52	22.98
9	41.55	21.01	35.56	26.32	21.92	17.79	52.19	12.49	0.00	12.49	52.19	25.42
Avg										9.06	43.34	21.80

Table 4: Interclass distance using normal TFIDF vectors

Class	1	2	3	4	5	6	7	8	9	Min	Max	Avg
1	0.00	38.81	11.27	27.09	36.59	50.78	18.23	51.18	50.73	11.27	51.18	31.63
2	38.81	0.00	28.90	13.00	3.61	13.54	55.40	18.12	21.36	3.61	55.40	21.42
3	11.27	28.90	0.00	17.15	26.69	40.90	28.02	41.75	41.77	11.27	41.77	26.27
4	27.09	13.00	17.15	0.00	10.73	25.14	43.76	27.23	28.56	10.73	43.76	21.41
5	36.59	3.61	26.69	10.73	0.00	15.69	53.21	19.62	22.45	3.61	53.21	20.96
6	50.78	13.54	40.90	25.14	15.69	0.00	67.34	13.39	19.02	13.39	67.34	27.31
7	18.23	55.40	28.02	43.76	53.21	67.34	0.00	67.27	66.34	18.23	67.34	44.40
8	51.18	18.12	41.75	27.23	19.62	13.39	67.27	0.00	13.01	13.01	67.27	27.95
9	50.73	21.36	41.77	28.56	22.45	19.02	66.34	13.01	0.00	13.01	66.34	29.25
Avg										10.90	57.07	27.84

Table 5: Interclass distance using extended TFIDF vectors

9 Conclusion

In this paper we investigated the importance of using NLP for improving Information Retrieval (IR). We identified document representation as a crucial step in an IR system and found that *incorrectness* and *insufficiency* of information are the major problems with traditional methods of document representation. We proposed to use Link Grammar and some other heuristics for addressing the problem of insufficiency of the information. We showed how to extend TFIDF vectors with the help of Self-Organizing Map (SOM). In order to find how *good* these document vectors were constructed, we used their information content as a measure. Since it is shown by Rong Jin *et al.* in SIGIR 2001 [8] with very large scale experiments that this measure is highly correlated with the precision-recall measurements in a typical IR system, we concluded that our proposed methods do aid in improving IR. In addition to this, we also found the interclass distances for TFIDF as well as our method and further provided the support for our proposed method. We are working toward applying our method for many other corpora and IR applications.

References

[1] E. Agirre and G. Rigau. Word sense disambiguation using conceptual density, 1996.

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

[3] 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.

[4] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[5] Simon Haykin. *Neural Networks: A Comprehensive Foundation*, chapter Self-Organizing Maps. Prentice Hall International, Inc., 1995.

[6] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen. Newsgroup exploration with WEBSOM method and browsing interface. Technical Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, January 1996.

[7] Bowen Hui. Applying NLP to IR: Why and How. Technical report, Dept. of CS, University of Toronto, April 1998.

[8] 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.

[9] 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.

[10] 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.

[11] T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1995.

[12] 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.

[13] B. P. Lathi. *An Introduction to Random Signals and Communication Theory*. International Textbook Company, Scanton, Pennsylvania, 1968.

[14] R. Mihalcea and D. Moldovan. A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, June 1999.

[15] Rada Mihalcea and Dan I. Moldovan. Word sense disambiguation based on semantic density. In Sanda Harabagiu, editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 16–22. Association for Computational Linguistics, Somerset, New Jersey, 1998.

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

[17] C. Monz. Computational semantics and information retrieval. In *J. Bos and M. Kohlhase (eds.) Proceedings of the 2nd Workshop on Inference in Computational Semantics (ICoS-2), 2000*, pages 1–5, 2000.

[18] 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.

[19] H. Ritter and T. Kohonen. Self-Organizing Semantic Maps. *Biological Cybernetics*, 61:241–254, 1989.

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

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

[22] M. Sanderson. Word Sense Disambiguation and Information Retreival. In *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieaval*, pages 142–151. Springer-Verlag, 1994.

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

[24] Daniel Sleator and Davy Temperley. Parsing english with a link grammar. In *Third International Workshop on Parsing Technologies*, August 1993.

[25] 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.

[26] 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.

[27] B. Yegnanarayana. *Artificial Neural Networds*, chapter 6, pages 201–232. Prentice Hall of India, 1999.