Using Linguistic Information to Improve the Performance of Vector-Based Semantic Analysis

Magnus Sahlgren (mange@sics.se)
David Swanberg (davidswanberg@yahoo.com)
RWCP Theoretical Foundation SICS Laboratory
Swedish Institute of Computer Science (SICS), Box 1263, SE-164 29 Kista, Sweden

Abstract

In this paper, we will show that the performance of vector-based semantic analysis can be improved by considering basic linguistic structures in the data—e.g. morphology. For this purpose, we have used a new method for vector-based semantic analysis that computes semantic word vectors based on distributed representations by means of random labeling of words in narrow context windows. This form of representation is more natural than previously reported techniques, and, as we will show, equivalent or even superior in performance when subjected to a standardized synonym test.

Vector-Based Semantic Analysis

The use of vector-based models of information for the purpose of semantic analysis is an area of research that has gained substantial recognition over the last decade. Pioneering techniques such as Latent Semantic Analysis (LSA; Landauer & Dumais, 1997) and Hyperspace Analogue to Language (HAL; Lund & Burgess, 1996) have demonstrated the viability of computing semantic word vectors from the co-occurrence statistics of words in large text data.

However, the prevailing techniques have been almost exclusively statistical, and consequently paid little or no attention to the linguistic structures of the data used in the experiments. This negligence regarding linguistics has, of course, been at least partly deliberate, as one of the primary goals of the techniques has been to develop representations of word meanings from text data "that was minimally preprocessed, not unlike human human-concept acquisition" (Burgess & Lund, 1998).

LSA and HAL are both purely statistical methods that treat the text data simply as a bag-of-words in which the only relevant piece of structural information is the words-by-contexts co-occurrence frequencies. What separates the two approaches is their treatment and conception of *context*. In LSA, the text data is represented as a words-by-documents co-occurrence matrix where each cell indicates the frequency of a given word in a given text sample of approximately 150 words. The frequencies are normalized, and the normalized matrix is transformed with Singular Value Decomposition (SVD) into a smaller matrix with reduced dimensionality. The purpose of using SVD to reduce the dimensions of the normalized frequency matrix is that this operation appears to accomplish inductive effects that capture latent semantic structures in the text data. Words are thus represented in the reduced matrix by semantic vectors of n dimensionality (300 proving to be optimal in Landauer & Dumais' (1997) experiments).

In HAL, the data is represented as a words-by-words co-occurrence matrix where each cell indicates the co-occurrence counts for a single word pair (a word pair being an asymmetrical relation so that "*xy*" and "*yx*" represent different entries in the matrix). Each word is thus represented in the matrix by both a row and a column, and these row/column pairs may be concatenated to produce a co-occurrence vector for each word. Assuming an $n \times n$ co-occurrence matrix, words are thus represented as semantic vectors of $2n$ dimensionality.

The point in all of this is that the word vectors capture relative meaning, thereby deserving the epithet "semantic". The semantic content is relative rather than absolute since it is only in relation to each other that the vectors *mean* anything, so semantic similarity between words can be established by comparing the vectors with each other. That the vectors in this way capture word meaning has been verified in a number of experiments where the high-dimensional vectors are used for executing different kinds of linguistic tasks pertaining to semantic knowledge, such as passing a standardized synonym test (Landauer & Dumais, 1997), comparing vector similarities with reaction times from lexical priming studies (Lund & Burgess, 1996), or evaluating the quality of content of student essays on given topics (Landauer, Laham, Rehder & Schreiner, 1997).

Random Indexing

We have studied the use of high-dimensional random distributed representations for accumulating a words-by-contexts co-occurrence matrix from which semantic word vectors can be extracted. In Kanerva, Kristofersson & Holst (2000), 1,800-dimensional semantic word vectors were computed using 1,800-dimensional sparse random *index vectors* representing documents of approximately 150 words each. The index vectors were accumulated into a words-by-contexts matrix by adding a document's index vector to the row for

a given word every time the word appeared in that document. This Random Indexing method is comparable to LSA except that the resulting matrix is significantly smaller than the words-by-documents matrix of LSA, since the dimensionality of the index vectors is smaller than the number of documents. By comparison, assuming a vocabulary of 60,000 words divided into 30,000 text samples, LSA would represent the data in a 60,000 × 30,000 words-by-documents matrix, whereas the matrix in Random Indexing would be 60,000 × 1,800, when 1,800-dimensional index vectors are used. This seems to accomplish the same inductive effects as those attained in LSA by applying SVD to the matrix, but in a more efficient way.

In the present experiment, the high-dimensional random vectors of Random Indexing have been used to index words and to calculate semantic word vectors by means of *narrow* context windows consisting of only a few adjacent words on each side of the focus word. As an example, imagine that the number of adjacent words in the context window is set to two. This would imply a window size of five space-separated linguistic units, i.e. the focus word and the two words preceding and succeeding it—a what we may call a "2 + 2 sized" context window. Thus, the context for the word *is* in the sentence *This parrot is no more* would be "This parrot" and "no more," as denoted by:

[(This parrot) is (no more)]

The reason for using narrow context windows as opposed to whole documents is the assumption that the semantically most significant context is the immediate vicinity of a word. Computing semantic word vectors using random indexing of words in narrow context windows is done by first assigning an *n*-dimensional sparse random vector called a *random label* to each word type in the text data. These random labels have a small number k of randomly distributed –1s and +1s, with the rest set to 0. The present experiment has utilized 1,800-dimensional random labels with $k = 8.7$ (±2.9). Thus, a label might have, for example, four –1s and five +1s.

Next, every time a given word—the focus word f_n—occurs in the text data, the labels for the words in its context window are added to its *context vector*. For example, assuming a 2 + 2 sized context window as represented by:

$$[(w_{n-2}\ w_{n-1})\ f_n\ (w_{n+1}\ w_{n+2})]$$

the context vector of f_n would be updated with:

$$L(w_{n-2}) + L(w_{n-1}) + L(w_{n+1}) + L(w_{n+2})$$

where $L(x)$ is the random label of x. This summation has also been weighted to reflect the distance of the words to the focus word. The weights were distributed so that the words immediately preceding and succeeding the focus word would get more significance in the computation of the context vectors. For the four different window sizes used in these experiments, the window slots were given weights as follows:

1 + 1: [(1) 0 (1)]
2 + 2: [(0.5, 1) 0 (1, 0.5)]
3 + 3: [(0.25, 0.5, 1) 0 (1, 0.5, 0.25)]
4 + 4: [(0.1, 0.1, 0.1, 1) 0 (1, 0.1, 0.1, 0.1)]

where the 0 in the middle represents the focus word.

This method is comparable to HAL, except that we use *distributed* representations that are more "brainlike," efficient and scalable. By comparison, assuming a vocabulary of 60,000 words, the HAL vectors would be (2 × 60,000) 120,000-dimensional, whereas our vectors are only 1,800-dimensional, regardless of the size of the vocabulary. Also, we use somewhat smaller context windows to capture the meaning of words. For example, in Burgess & Lund (1998), a context window spanning 10 words were used, whereas we found in our experiments that the performance of the method degrades significantly when the window size exceeds an upper limit of 4 + 4 words.

Introducing Linguistic Information

Up to this point, the only structural relations of language that are utilized in the creation of the context vectors by the above-described techniques are the distributional patterns of linguistic entities. Since there are more complex structural features of language (like morphology and part-of-speech information, e.g.) that may very well be significant for uncovering semantic information, it seems unmotivated not to take these features into account. By introducing linguistic information to the system, we would have the opportunity to investigate whether or not this addition of structural information enhances the performance of vector-based semantic analysis.

First, a naïve form of morphological analysis was tested. By simply truncating the word tokens at a predefined number of letters one would hope to approximate word stems. Truncation lengths of 6, 8, 10 and 12 were investigated. A more reliable and linguistically established way of extracting the word stems of each word token is, of course, to use a parser. In our experiments, the Conexor FDG-parser was used. The point in using morphological analysis is to convert word tokens into word types, thus reducing the vocabulary and thereby compressing the words-by-context matrix. By comparison, the complete vocabulary when no preprocessing has occurred is, in our text data, 94,000 words, and when truncating the words after eight characters 79,000.

In an attempt to resolve problems due to ambiguity, the morphologically analyzed text was also marked with part-of-speech information. Since the system (without linguistic information) is insensitive to different semantic meaning with two or more identical word representations (e.g. the verb and the noun "roll"), it will create the exact

same entry in the matrix for them both. By appending to the beginning of every word a part-of-speech tag consisting of one letter, this ambiguity will be remedied. In this way the verb "roll" would be "vroll" and the respective noun would be "nroll," allowing the system to produce different entries in the matrix for the originally same word representations depending on which part-of-speech the representation in question affiliates to.

Evaluation and Results

The technique was evaluated on a ten-million-word balanced corpus of unmarked English with the help of a vocabulary test, TOEFL (Test Of English as a Foreign Language). This is a standardized test employed by, for example, American universities to survey foreign applicants' knowledge of the English language. In the synonym finding part of the test, the test taker is asked to find the synonyms to certain given words. For each given word, a multiple-choice answering suggestion of four alternatives is provided, where one alternative is the intended synonym, and is supposed to be indicated by the test person. In the present experiment, 80 test items of this type have been used.

Table 1: Average results (±1.5) given in percent of correct answers to the TOEFL-test, where Tr. means truncation length, WS means word stems and PoS+WS means part-of-speech tagged word stems.

Linguistic Analysis	Context Window				Average (±0.73)
	1 + 1	2 + 2	3 + 3	4 + 4	
None	64.5	67	65.3	65.5	65.6
Tr. 6	55	57.5	57.3	55.3	56.3
Tr. 8	61.5	64.3	62	63.3	62.8
Tr. 10	66	68.5	66.3	66.3	66.8
Tr. 12	64.8	65.3	63.8	64.8	64.6
WS	63.5	70.8	72	66	68.1
PoS+WS	66	64.5	65	65.5	65.3
Average (±0.56)	63.0	65.4	64.5	63.8	

The numbers in the cells of Table 1 are the average results of five runs. The standard deviation for these results is 1.5. For the average result of each context window, the standard deviation is 0.56, and for the average of each "linguistic parameter" 0.73. All results are given in percent of correct answers to the TOEFL-test. By comparison, tests with LSA on the same text data, using the LSIBIN program from Telecordia Technologies, produced top scores at 600 factors of 58.75% using the unnormalized matrix, and 65% using a normalized one. The average result reported by Landauer & Dumais (1997) with LSA (using normalization and different text data) is 64.4%, and foreign (non-English speaking) applicants to U.S. colleges average 64.5%.

The results from our experiments show that by using high-dimensional random distributed representations to label words in narrow context windows, it is possible to reach a result on a standardized synonym test that is equivalent with the performance of previously reported techniques. Without using linguistic information, the system averages 65.6%. However, when supplying morphological information in the form of carefully applied truncation (using a truncation length of 10 characters), the system's average result increases to 66.8% correct answers. When using stemming of the words, the result is even better, with an average of 68.1% correct answers to the synonym part of TOEFL.

Adding part-of-speech information does not further improve the performance reached when using carefully applied truncation or proper word-stem analysis. The average result when adding part-of-speech information drops to 65.3%. This might be an effect of the increase in the number of unique words in the text data that is the consequence of supplying part-of-speech information for each word.

That the inclusion of morphology in the form of proper word-stem analysis or carefully applied truncation yields the best overall results indicates that taking advantage of other inherent structural relations in text, in addition to the distributional patterns of linguistic entities, really might be significant for uncovering semantic information. We thus conclude that the performance of vector-based semantic analysis benefits from the implementation of linguistic information.

Acknowledgements

We wish to thank Pentti Kanerva, Anders Holst and Jussi Karlgren. This research is funded by Japan's Ministry of International Trade and Industry (MITI) through Real World Computing Partnership (RWCP). The training corpus and 80 TOEFL-test items used in these experiments were provided by courtesy of Professor Thomas Landauer, University of Colorado.

References

Burgess, C. & Lund, K. (1998) The dynamics of meaning in memory. In Dietrich, E. & Markman, A. B. (Eds.), (2000) *Cognitive dynamics: Conceptual change in humans and machines*. Mahwah, NJ: Lawrence Erlbaum Associates.

Kanerva, P., Kristofersson, J. & Holst, A. (2000) Random Indexing of text samples for Latent Semantic Analysis. In Gleitman, L.R. & Josh, A.K. (Eds.), *Proceedings of the 22nd Annual Conference of the Cognitive Science Society* (p. 1036). Mahwah, New Jersey: Erlbaum.

Landauer, T. K. & Dumais, S. T. (1997) A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition, induction and

representation of knowledge. *Psychological Review*, 104 (2), pp. 211–240.

Landauer, T. K., Laham, D., Rehder, B., & Schreiner, M. E. (1997) How well can passage meaning be derived without using word order? A comparison of latent semantic analysis and humans. In Shafto, M. G. & Langley, P. (Eds.), *Proceedings of the 19th annual meeting of the Cognitive Science Society* (pp. 412–417). Mawhwah, NJ: Erlbaum.

Lund, K. & Burgess, C. (1996) Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers*, 28 (2), pp. 203–208.