# Global Evaluation of Random Indexing through Swedish Word Clustering Compared to the People's Dictionary of Synonyms

Magnus Rosell
KTH CSC
Stockholm, Sweden
*rosell@csc.kth.se*

Martin Hassel
DSV, KTH - Stockholm University
Kista, Sweden
*xmartin@dsv.su.se*

Viggo Kann
KTH CSC
Stockholm, Sweden
*viggo@nada.kth.se*

**Abstract**

Evaluation of word space models is usually local in the sense that it only considers words that are deemed very similar by the model. We propose a global evaluation scheme based on clustering of the words. A clustering of high quality in an external evaluation against a semantic resource, such as a dictionary of synonyms, indicates a word space model of high quality.

We use Random Indexing to create several different models and compare them by clustering evaluation against the People's Dictionary of Synonyms, a list of Swedish synonyms that are graded by the public. Most notably we get better results for models based on syntagmatic information (words that appear together) than for models based on paradigmatic information (words that appear in similar contexts). This is quite contrary to previous results that have been presented for local evaluation.

Clusterings to ten clusters result in a recall of 83% for a syntagmatic model, compared to 34% for a comparable paradigmatic model, and 10% for a random partition.

**Keywords**

Random Indexing, Word Space Model, Word Clustering, Evaluation, Dictionary of Synonyms

## 1 Introduction

Word space models (see among others [1, 16, 11, 6, 15]) map words to vectors in a multidimensional space by extracting statistics about the context they appear in from a large sample of text. Words that thus become represented by similar vectors (as measured by a similarity measure such as the cosine measure) are considered related. What this (meaning) relation could be referred to in ordinary (human) semantics is not obvious. It may capture something like synonymy, but may as well regard for instance antonyms, and a hyponym and its hyperonym as highly related.

Relations between words based on their contexts can be divided into two categories [15]: Two words have a relation that is

*syntagmatic* if they appear together.
*paradigmatic* if they appear in similar contexts.

Word space models can be constructed in attempts to capture either of these two relations. In this work we use Random Indexing (see Section 2) to construct several different word space models.

Word space models have been evaluated using several different schemes [15]. They are all *local* in that they only consider a small part of the words in the model. We introduce a new *global* evaluation scheme that takes all words in the model into consideration, using word clustering and a list of synonyms.

The paper is organized as follows. Sections 2 and 3 describe Random Indexing and word clustering. We discuss evaluation of word space models in general and present our proposed global evaluation scheme in Section 4. In Section 5 we describe and discuss our experiments: the text set we have used (Section 5.1) and evaluation against a list of Swedish synonyms, called the People's Dictionary of Synonyms (Section 5.2). Finally, Section 6 contains some conclusions.

## 2 Random Indexing

Random Indexing (RI) [6, 13] is an efficient and scalable implementation of the word space model idea. It can be used for attempts at capturing both syntagmatic and paradigmatic relations, and has been shown to perform on par with other implementations. In the paradigmatic version RI assigns a sparse *random vector* to each word, usually with a dimension of a few thousands, say $n$. The random vectors only contain $2t$ ($t \ll n$) randomly chosen non-zero elements, half of which are assigned one (1), and half minus one (-1).

The random vectors are used to construct *context vectors* for all words. The method runs through the texts word by word focusing on a center word. A portion of the surrounding words are considered being in a *sliding window*. We have used symmetric windows with $\omega$ words on both sides of the center word included. As the sliding window moves through the text the random vectors of the surrounding words are added to the context vector of the the current center word. The addition may be either constant or weighted depending on the distance, $d$, between the center word and the particular surrounding word. We have used constant weighting and the commonly used exponential dampening: $2^{1-d}$. The resulting word vectors will be similar for words that appear in similar contexts. We measure the similarity/relatedness between two words by the cosine similarity of their corresponding context

vectors (the dot product of the normalized vectors)[1].

In the syntagmatic version of RI random vectors are assigned to each text. If a word appears in a text the random vector of the text is added to the context vector of the word[2]. We define the similarity between two words as in the paradigmatic version. It now measures to what extent the words appear in the same texts.

Although, being reasonable approximations of syntagmatic and paradigmatic relations the two RI versions are closely related, as noted in [15]. Consider the constant weighting function for the paradigmatic version. If we increase $\omega$ until it covers whole texts each word in the text is updated with the sum of all the random vectors in the text (except the one associated with itself, a very small part of the sum for large enough texts). This sum serves as a "random vector" (albeit not sparse) for the text, which means that we have a method that is similar to the syntagmatic version[3]. These dense "random vectors" become similar if the texts share a lot of words. In such cases the paradigmatic model is prevented from being fully transformed into a syntagmatic one. However, if the syntagmatic model performs better than a corresponding paradigmatic one, we conjecture that the latter will gain from having its sliding window increased.

# 3 Word Clustering

We use the K-Means clustering algorithm (see for instance [12]) to cluster the words based on the word space models. K-Means improves on $k$ centroids (component-wise average vectors), that represent $k$ clusters, by iteratively assigning words to the cluster with the most similar centroid. We have set 20 iterations as maximum, as the quality of clustering usually improves most at the beginning of the process.

We use the dot product for similarity between the normalized word vectors and the centroids, i.e. the average cosine similarity between the word and all words in a cluster. In each iteration all words are compared to all centroids, meaning that when a word is assigned to a cluster all other words are taken into consideration. This is an appealing property of the algorithm in its own right. It also makes it suitable for the evaluation scheme we present in the next section.

# 4 Evaluation

Word space models have been evaluated using several different resources and evaluation metrics [15]. In [14] evaluation methods are divided into two categories: *indirect* schemes evaluate a word space model through an application and are therefore not concerned with the model per se, while *direct* schemes compare a model to some lexical resource, to judge its ability to model the information it contains.

The existing evaluation schemes are *local* – they only consider a small part of the words in the model. The most common direct evaluation scheme is to use a synonym test: for each question the model is considered successful if the similarity of the test word to the correct alternative is higher than to the other. Here, only the words in the synonym test are regarded. How they relate to the other words is not taken into consideration. In fact, it is only the words within the same question that are considered at the same time.

## 4.1 Global Evaluation

The *global* evaluation scheme we propose takes the relation between all words of the model into account. We cluster all words represented in a model; all words are assigned to one of several clusters by means of the similarity measure. In the assignment of each word all other words are considered via the clusters they appear in. This is true for most clustering algorithms, and in particular for the K-Means algorithm, see Section 3.

> *The global evaluation scheme considers a word space model to be of high quality if it leads to clusterings of high quality. This quality reflects how all the words relate to each other.*

When the clustering evaluation is performed using a lexical resource (such as a list of synonyms), we have a global and direct word space model evaluation. There are many measures of clustering quality that could be used to compare the models. The next section discusses word clustering evaluation, in particular the evaluation measures appropriate for our experiments.

In [8] it is argued that the most interesting information of a word space model is found in the local structure, rather than in the global. This should not be confused with our global evaluation. It is the local relations (similarities between words) that drives the clustering; it takes *all* local relations into consideration. Further, when the evaluation is made against a lexical resource, it concerns the local structure (there are few synonyms to each word compared to the number of words in the model).

## 4.2 Word Clustering Evaluation

Clustering evaluation can be internal or external. We are interested in how the underlying word space model relation compares to what words humans consider related; i.e. we want to compare the clustering result to a resource through external evaluation. Depending on the resource this could be achieved in several ways.

In the following experiments (Section 5) we compare the results to a synonym dictionary that consists of pairs of synonyms (Section 5.2). There are several measures (see for instance [12] and [4]), that compare a clustering to a known categorization based on pairs of words. Each pair can be either in the same or in two different clusters, and in the same category or not. This gives us the four counts presented in the left part of Table 1: $tp$ is for true positives, the number of pairs of words that appear in the same cluster *and* in the same category, $fp$, $fn$, and $tn$ are for false positives,

[1] The method corresponds to a projection of the words represented in a space defined by the ordinary word-word-co-occurrence-matrix to a random subspace. When the original data matrix is sparse and the projection is constructed well the distortions in the similarities are small [9].

[2] This results in a random matrix projection of the common term-by-document matrix used in search engines.

[3] For the paradigmatic RI version with a weighting function that decreases with the distance $d$ this relatedness is not as strong, but could perhaps be of some significance.

| Cluster | Category | | In/not in dictionary |
|---|---|---|---|
| | Same | Different | |
| Same | $tp$ | $fp$ | $tp$ |
| Different | $fn$ | $tn$ | $fn$ |

**Table 1:** *Number of Pairs in the Same and Different Clusters, and in a Categorization or a Dictionary*

false negatives, and true negatives. Using these several measures can be constructed, the most straightforward perhaps precision ($p$) and recall ($r$): $p = \frac{tp}{tp+fp}$, $r = \frac{tp}{tp+fn}$. These measures depend on that we know a full categorization, which is not the case in our experiments; pairs that are not in the synonym dictionary may still be synonymous or have some other relation. We do not know what these relations might be, so we can not use the pairs not in the dictionary.

The only counts we can define using a dictionary of synonyms are the ones in the right part of Table 1. Hence, the only measure we can define is recall, $r$. It denotes the part of the synonym pairs that appear in the same cluster. It is important to note that a high recall does not necessarily imply that most of the words in a cluster are related, only that the synonym pairs are not split between clusters.

To put the evaluation in perspective we present the results for random partitions as well as the results for the clustering algorithm applied on the different models. In a random partition with $k$ parts (clusters), for each word in a pair the probability for the other word of being in the same cluster is $1/k$. Thus the recall for the entire random partition is $1/k$. The clustering result, of course, has to outperform the random partition to be considered any good at all.

### 4.3 Local Evaluation via Clustering

If we cluster just the words that also appear in the resource we compare the clustering to, we make a *local* evaluation, which is much more similar to previously used schemes. It does, however, consider the relations between all the words in the resource. This is usually not the case for other local schemes, as described for the synonym test previously.

## 5 Experiments

We have clustered the words based on several different RI models, that we constructed using a freely available tool-kit called JavaSDM[4]. In all models we have used eight non-zero elements in the random vectors ($t = 4$). We use the following notation to abbreviate differences between the models, see Section 2: "$n$-win$\omega$", or "$n$-text". win$\omega$ means a sliding window with $\omega$ words before and after the center word, text means that we have used texts as contexts, and $n$ is the dimension of the vectors. We have used the exponential dampening weighting function for the $n$-win$\omega$-methods. We indicate constant weighting thus: "$n$-win$\omega$-const".

[4] http://www.nada.kth.se/∼xmartin/java/JavaSDM/

As K-Means is not deterministic we cluster the words ten times for each representation and calculate averages and standard deviations. We can only compare results for the same number of clusters. For two results to be considered different they, as a rule of thumb, must not overlap with the standard deviations.

### 5.1 Text Set

The RI:s have been trained on a text set consisting of all texts from the Swedish Parole corpus [3], 20 million words, the Stockholm-Umeå Corpus [2], 1 million words, and the KTH News Corpus [5], 18 million words. In all they contain 114 691 files/texts. We tokenized and lemmatized all texts using GTA, the Granska Text Analyzer [10], removed stop words (function words and extremely frequent words) and all words that appeared less than four times.

### 5.2 People's Dictionary of Synonyms

For the evaluation we have used the People's Dictionary of Synonyms [7], a dictionary produced by the public. In 2005 a list of possible synonyms was created by translating all Swedish words in a Swedish-English dictionary to English and then back again using an English-Swedish dictionary. The generated pairs contained lots of non-synonyms. The worst pairs were automatically removed using Random Indexing.

Every user of the popular dictionary Lexin online was given a randomly chosen pair from the list, and asked to judge it. An example (translated from Swedish): "*Are 'spread' and 'lengthen' synonyms?* Answer using a scale from 0 to 5 where 0 means *I don't agree* and 5 means *I do fully agree*, or answer *I do not know*." Users of the dictionary could also propose pairs of synonyms, which subsequently were presented to other users for judgment.

All responses were analyzed and screened for spam. The good pairs were compiled into the dictionary. Millions of contributions have resulted in a constantly growing dictionary of more than 75 000 Swedish pairs of synonyms. Since it is constructed in a giant cooperative project, the dictionary is a free downloadable language resource[5].

An interesting feature of the People's Dictionary of Synonyms is that the synonymity of each pair is graded. It is the mean grading by the users who have judged the pair. The available list contains 18 053 pairs that have a grading of 3.0 to 5.0 in increments of 0.1. Through the rest of the paper we refer to this part of the dictionary as *Synlex*. (See Table 4 and our complementing paper[6].)

### 5.3 Results

The results in Table 2 follow the global evaluation scheme of Section 4.1, while Table 3 uses the local scheme presented in Section 4.3. Where the standard deviation is 0.00 for the random partitions[7] we have

[5] http://lexin.nada.kth.se/synlex

[6] http://www.csc.kth.se/ rosell/publications/papers/rosellkannhassel09complement.pdf

[7] This is the case for large enough sets of words.

| $k$ | Representation dim-context(-const) | Recall (stdv) K-Means | Random |
|---|---|---|---|
| **100** | **1800-text** | 0.56 (0.10) | 0.01 |
| 100 | 1800-win4 | 0.15 (0.01) | 0.01 |
| 5 | 500-text | 0.48 (0.12) | 0.20 |
| **5** | **1000-text** | 0.77 (0.07) | 0.20 |
| **5** | **1800-text** | 0.83 (0.01) | 0.20 |
| **10** | **500-text** | 0.77 (0.05) | 0.10 |
| **10** | **1000-text** | 0.80 (0.05) | 0.10 |
| **10** | **1800-text** | 0.83 (0.02) | 0.10 |
| 5 | 500-win4 | 0.41 (0.02) | 0.20 |
| 5 | 1000-win4 | 0.42 (0.02) | 0.20 |
| 5 | 1800-win4 | 0.44 (0.03) | 0.20 |
| 10 | 500-win4 | 0.32 (0.01) | 0.10 |
| 10 | 1000-win4 | 0.31 (0.01) | 0.10 |
| 10 | 1800-win4 | 0.34 (0.02) | 0.10 |
| 5 | 500-win30 | 0.44 (0.03) | 0.20 |
| 5 | 1000-win30 | 0.43 (0.03) | 0.20 |
| 5 | 1800-win30 | 0.45 (0.03) | 0.20 |
| 10 | 500-win30 | 0.34 (0.03) | 0.10 |
| 10 | 1000-win30 | 0.34 (0.01) | 0.10 |
| 10 | 1800-win30 | 0.33 (0.01) | 0.10 |
| 5 | 500-win250 | 0.42 (0.02) | 0.20 |
| 5 | 1000-win250 | 0.41 (0.03) | 0.20 |
| 5 | 1800-win250 | 0.44 (0.02) | 0.20 |
| 10 | 500-win250 | 0.33 (0.01) | 0.10 |
| 10 | 1000-win250 | 0.34 (0.02) | 0.10 |
| 10 | 1800-win250 | 0.33 (0.01) | 0.10 |
| 5 | 500-win30-const | 0.45 (0.03) | 0.20 |
| 5 | 1000-win30-const | 0.43 (0.02) | 0.20 |
| 5 | 1800-win30-const | 0.44 (0.03) | 0.20 |
| 10 | 500-win30-const | 0.34 (0.02) | 0.10 |
| 10 | 1000-win30-const | 0.34 (0.02) | 0.10 |
| 10 | 1800-win30-const | 0.34 (0.01) | 0.10 |
| **5** | **500-win250-const** | 0.72 (0.07) | 0.20 |
| 5 | 1000-win250-const | 0.66 (0.04) | 0.20 |
| 5 | 1800-win250-const | 0.76 (0.09) | 0.20 |
| 10 | 500-win250-const | 0.58 (0.03) | 0.10 |
| 10 | 1000-win250-const | 0.56 (0.03) | 0.10 |
| 10 | 1800-win250-const | 0.60 (0.01) | 0.10 |
| **5** | **500-win1000-const** | 0.67 (0.04) | 0.20 |
| 5 | 1000-win1000-const | 0.68 (0.05) | 0.20 |
| 5 | 1800-win1000-const | 0.69 (0.06) | 0.20 |
| 10 | 500-win1000-const | 0.58 (0.02) | 0.10 |
| 10 | 1000-win1000-const | 0.60 (0.03) | 0.10 |
| 10 | 1800-win1000-const | 0.60 (0.03) | 0.10 |

**Table 2:** Global Evaluation. *The Effect of Different Contexts. Recall for Word Clustering of All Words, RI in Table 4. (k – the number of clusters) The table is divided into four sections by horizontal double lines. The top one contains results for clusterings to 100 clusters. The second one contains the results for the syntagmatic models, and the two following the results for the paradigmatic models with two different weightings: those with the exponential damping and those with constant (-const). The best representation for each number of clusters is presented in bold face letters (for ties: both). The standard deviation for the random "clustering" is 0.00 in all cases.*

not reported it. The best representation for each number of clusters is presented in bold face letters. For ties, i.e. results with overlapping standard deviations, we present them both with bold face letters.

We present the number of words and pairs in Synlex and the RI:s in Table 4. See also our complementing paper[6]. The pairs in Synlex that are not in the RI are mostly multi-word tokens, words in non lemma form, and slang words that the public has wanted to include.

| $k$ | Representation dim-context | Recall (stdv) K-Means | Random |
|---|---|---|---|
| 5 | 500-text | 0.22 (0.01) | 0.20 |
| 5 | 1000-text | 0.30 (0.03) | 0.20 |
| **5** | **1800-text** | 0.45 (0.06) | 0.20 |
| 10 | 500-text | 0.11 (0.00) | 0.10 |
| 10 | 1000-text | 0.19 (0.04) | 0.10 |
| **10** | **1800-text** | 0.31 (0.08) | 0.10 |
| **5** | **500-win4** | 0.39 (0.00) | 0.20 |
| **5** | **1000-win4** | 0.40 (0.01) | 0.20 |
| **5** | **1800-win4** | 0.40 (0.00) | 0.20 |
| **10** | **500-win4** | 0.27 (0.01) | 0.10 |
| **10** | **1000-win4** | 0.27 (0.02) | 0.10 |
| **10** | **1800-win4** | 0.28 (0.01) | 0.10 |

**Table 3:** Local Evaluation. *Recall for Word Clustering of Words Only in Synlex, Synlex*RI in Table 4. (k – the number of clusters) The best representation for each number of clusters is presented in bold face letters (for ties: both). The standard deviation for the random "clustering" is 0.00 in all cases.*

## 5.4 Discussion

Our major finding is that the syntagmatic RI versions perform much better than the paradigmatic versions in our global evaluation. This is apparent in Table 2, which contains the results for the syntagmatic versions ("$n$-text") and several paradigmatic versions. This result differ to local direct evaluations that have been performed against synonym resources, where paradigmatic versions have been more successful [15].

This, present, result may seem counterintuitive, as synonyms have a paradigmatic relation. A plausible explanation is that for the syntagmatic versions the cluster centroids actually capture something very similar to paradigmatic relations. Consider a clustering of the words represented in the the term-by-document matrix that the syntagmatic RI model is an approximation of (see Section 3). Synonyms usually appear with a set of shared words. These words will be likely to be assigned to the same cluster as they often appear together. As the synonyms also appear with them chances are that they also will end up in that cluster. The centroid associates synonyms via the words they both appear together with – a paradigmatic relation extracted from a syntagmatic representation.

The paradigmatic RI models are approximations of the word-word-cooccurence matrix (see Section 3) that contains the overall distribution of the close context of each word. It is a direct attempt at capturing the paradigmatic relations between words. However, the clustering can not find associations between words that appear further apart within specific documents. It is only for really large windows and the constant weighting scheme ("-const") a paradigmatic version can compete. This is in line with the argument in Section 2 that a paradigmatic version with large windows and constant weighting scheme is closely related to the syn-

| | Pairs | Words ($n$) | Possible Pairs ($n(n-1)/2$) |
|---|---|---|---|
| Synlex | 18 053 | 15 296 | $1.67 \cdot 10^8$ |
| RI | $9.43 \cdot 10^9$ | 137 364 | $9.43 \cdot 10^9$ |
| Synlex*RI | 14 051 | 11 173 | $6.24 \cdot 10^7$ |

**Table 4:** *Pairs and Words in Synlex and RI.* Synlex*RI *means pairs that appear in both Synlex and RI.*

tagmatic version. The paradigmatic version with constant weighting scheme improves with increasing window size ($2 \cdot \omega$), but seems to be saturated at $\omega = 250$, since results do not improve for $\omega = 1000$. A window size of 500 covers most texts in their entire. That the paradigmatic versions with exponential weighting (not "-const") does not improve with increasing window size is not surprising; the impact of words far away from the center word is limited.

The syntagmatic versions perform better with increasing dimensionality ($n$). This suggests that they might benefit more from even larger dimensionality. The paradigmatic versions are not effected.

The results for the local evaluation (see Section 4.3) in Table 3 gives a different view. The syntagmatic models perform much worse than in the global evaluation, while the paradigmatic models perform similarly. Here, the paradigmatic models outperform the syntagmatic models, for low dimensionalities. In fact, the syntagmatic model performs as a random partition for $n = 500$. However, as in the global evaluation the syntagmatic version performs better with increasing dimensionality. For $n = 1800$ it performs comparable to the syntagmatic version.

The syntagmatic model exploits the information in all of the words it contains and performs much better when it is allowed to use them (global vs. local evaluation). Then it outperforms the paradigmatic models. The results for the paradigmatic models are unaffected by whether they are allowed to consider all other words. Both versions obviously have their merits. We observe that the best performing of the evaluated models is 1800-text, the syntagmatic model with a dimension of $n = 1800$. It performs superior to all paradigmatic models in the global evaluation and comparable in the local evaluation. In the global evaluation, for ten clusters, it achieves 83% recall, compared to 34% for the paradigmatic models with exponential dampening, and 10% for the random partitions.

None of the models is able to separate the different Synlex gradings. We have confirmed this in two ways (see our complementing paper[6]): by plotting the distributions of gradings and model similarities, and by evaluating using only the synonym pairs of high grade (results were similar to Table 2). The models do, however, give higher similarity to synonyms in the dictionary than to other word pairs.

# 6 Conclusions

We have presented and used a new *global* evaluation scheme for word space models. While local evaluation only considers a portion of the words in the model, global evaluation takes them all into consideration.

We constructed word space models (realized using Random Indexing) on Swedish texts and used a list of synonyms called the The People's Dictionary of Synonyms for evaluation. In our global evaluation scheme models that attempt to capture syntagmatic relations between words performed better than models that attempt to capture paradigmatic relations. This result is contrary to previous results using local evaluation against synonym resources.

This work addresses the theoretic matter of how to evaluate word space models. Though we hope that the use of a combination of both local and global evaluation will promote the investigation of the nature of word space models and the word (meaning/similarity) relation they define, we conclude the paper with a more tangible question. The syntagmatic models perform very well when they are allowed to take all words into consideration. How can this be exploited in applications?

# References


[1] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *J. of the Society for Inform. Science*, 41(6):391–407, 1990.

[2] E. Ejerhed, G. Källgren, O. Wennstedt, and M. Åström. *SUC - The Stockholm-Umeå Corpus*, version 1.0 (suc 1.0). CD-ROM. The Dept of Linguistics, University of Stockholm and the Dept of Linguistics, University of Umeå. ISBN 91-7191-348-3, 1992.

[3] M. Gellerstam, Y. Cederholm, and T. Rasmark. The bank of Swedish. In *Proc. of Second Int. Conf. on Lang. Resources and Evaluation. LREC-2000*, Athens, Greece, 2000.

[4] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. *J. of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[5] M. Hassel. Automatic construction of a Swedish news corpus. In *Proc. 13th Nordic Conf. on Comp. Ling. – NODALIDA '01*, 2001.

[6] P. Kanerva, J. Kristofersson, and A. Holst. Random indexing of text samples for latent semantic analysis. In *Proc. of the 22nd annual conference of the cognitive science society*, 2000.

[7] V. Kann and M. Rosell. Free construction of a free Swedish dictionary of synonyms. In *Proc. 15th Nordic Conf. on Comp. Ling. – NODALIDA '05*, 2005.

[8] J. Karlgren, A. Holst, and M. Sahlgren. Filaments of meaning in word space. *Advances in Information Retrieval*, 2008.

[9] S. Kaski. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proc. of IJCNN'98, Int. Joint Conf. on Neural Networks*, volume 1. IEEE Service Center, Piscataway, NJ, 1998.

[10] O. Knutsson, J. Bigert, and V. Kann. A robust shallow parser for Swedish. In *Proc. 14th Nordic Conf. on Comp. Ling. – NODALIDA '03*, 2003.

[11] T. K. Landauer and S. T. Dumais. A solution to Plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240, 1997.

[12] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[13] M. Sahlgren. An introduction to random indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th Int. Conf. on Terminology and Knowledge Engineering, TKE 2005*, 2005.

[14] M. Sahlgren. Towards pertinent evaluation methodologies for word-space models. In *In Proc. of the 5th Int. Conf. on Lang. Resources and Evaluation*, Genoa, Italy, 2006.

[15] M. Sahlgren. *The Word-Space Model: using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD thesis, Stockholm University, 2006.

[16] H. Schütze. Word space. In S. Hanson, J. Cowan, and C. Giles, editors, *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann Publishers, 1993.