Random Indexing using Statistical Weight Functions

James Gorman and **James R. Curran**
School of Information Technologies
University of Sydney
NSW 2006, Australia
{jgorman2,james}@it.usyd.edu.au

Abstract

Random Indexing is a vector space technique that provides an efficient and scalable approximation to distributional similarity problems. We present experiments showing Random Indexing to be poor at handling large volumes of data and evaluate the use of weighting functions for improving the performance of Random Indexing. We find that Random Index is robust for small data sets, but performance degrades because of the influence high frequency attributes in large data sets. The use of appropriate weight functions improves this significantly.

1 Introduction

Synonymy relations between words have been used to inform many Natural Language Processing (NLP) tasks. While these relations can be extracted from manually created resources such as thesauri (e.g. Roget's Thesaurus) and lexical databases (e.g. WordNet, Fellbaum, 1998), it is often beneficial to extract these relationships from a corpus representative of the task.

Manually created resources are expensive and time-consuming to create, and tend to suffer from problems of bias, inconsistency, and limited coverage. These problems may result in an inappropriate vocabulary, where some terms are not present or an unbalanced set of synonyms. In a medical context it is more likely that administration will refer to the giving of medicine than to paper work, whereas in a business context the converse is more likely.

The most common method for automatically creating these resources uses distributional similarity and is based on the *distributional hypothesis* that *similar words appear in similar contexts*. Terms are described by collating information about their occurrence in a corpus into vectors. These *context vectors* are then compared for similarity. Existing approaches differ primarily in their definition of *context*, e.g. the surrounding words or the entire document, and their choice of distance metric for calculating similarity between the context vectors representing each term.

In this paper, we analyse the use of Random Indexing (Kanerva et al., 2000) for semantic similarity measurement. Random Indexing is an approximation technique proposed as an alternative to Latent Semantic Analysis (LSA, Landauer and Dumais, 1997). Random Indexing is more scalable and allows for the incremental learning of context information.

Curran and Moens (2002) found that dramatically increasing the volume of raw input data for distributional similarity tasks increases the accuracy of synonyms extracted. Random Indexing performs poorly on these volumes of data. Noting that in many NLP tasks, including distributional similarity, statistical weighting is used to improve performance, we modify the Random Indexing algorithm to allow for weighted contexts.

We test the performance of the original and our modified system using existing evaluation metrics. We further evaluate against bilingual lexicon extraction using distributional similarity (Sahlgren and Karlgren, 2005). The paper concludes with a more detailed analysis of Random Indexing in terms of both task and corpus composition. We find that Random Index is robust for small corpora, but larger corpora require that the contexts be weighted to maintain accuracy.

Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006), pages 457–464,
Sydney, July 2006. ©2006 Association for Computational Linguistics

2 Random Indexing

Random Indexing is an approximating technique proposed by Kanerva et al. (2000) as an alternative to Singular Value Decomposition (SVD) for Latent Semantic Analysis (LSA, Landauer and Dumais, 1997). In LSA, it is assumed that there is some underlying dimensionality in the data, so that the attributes of two or more terms that have similar meanings can be *folded* onto a single axis.

Sahlgren (2005) criticise LSA for being both computationally inefficient and requiring the formation of a full co-occurrence matrix and its decomposition before any similarity measurements can be made. Random Indexing avoids both these by creating a short *index vector* for each unique context, and producing the *context vector* for each term by summing index vectors for each context as it is read, allowing an incremental building of the context space.

Hecht-Nielsen (1994) observed that there are many more nearly orthogonal directions in high-dimensional space than there are truly orthogonal directions. The random index vectors are *nearly-orthogonal,* resulting in an approximate description of the context space. The approximation comes from the Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss, 1984), which states that if we project points in a vector space into a randomly selected subspace of sufficiently high dimensionality, the distances between the points are approximately preserved. Random Projection (Papadimitriou et al., 1998) and Random Mapping (Kaski, 1998) are similar techniques that use this lemma. Achlioptas (2001) showed that most zero-mean distributions with unit variance, including very simple ones like that used in Random Indexing, produce a mapping that satisfies the lemma. The following description of Random Indexing is taken from Sahlgren (2005) and Sahlgren and Karlgren (2005).

We allocate a d length index vector to each unique context as is it found. These vectors consist of a large number of 0s and a small number (ϵ) of ±1s. Each element is allocated one of these values with the following probability:

$$\begin{cases} +1 & \text{with probability} \frac{\epsilon/2}{d} \\ 0 & \text{with probability} \frac{d-\epsilon}{d} \\ -1 & \text{with probability} \frac{\epsilon/2}{d} \end{cases}$$

Context vectors are generated *on-the-fly*. As the corpus is scanned, for each term encountered, its contexts are extracted. For each new context, an index vector is produced for it as above. The context vector is the sum of the index vectors of all the contexts in which the term appears.

The context vector for a term t appearing in one each in the contexts $c_1 = [1, 0, 0, -1]$ and $c_2 = [0, 1, 0, -1]$ would be $[1, 1, 0, -2]$. If the context c_1 encountered again, no new index vector would be generated and the existing index vector for c_1 would be added to the existing context vector to produce a new context vector for t of $[2, 1, 0, -3]$.

The distance between these context vectors can then be measured using any vector space distance measure. Sahlgren and Karlgren (2005) use the cosine measure:

$$\cos(\theta(u,v)) = \frac{\vec{u} \cdot \vec{v}}{|\vec{u}|\ |\vec{v}|} = \frac{\sum_{i=1}^{d} \vec{u}_i \vec{v}_i}{\sqrt{\sum_{i=1}^{d} \vec{u}_i^2}\sqrt{\sum_{i=1}^{d} \vec{v}_i^2}}$$

Random Indexing allows for incremental sampling. This means that the entire data set need not be sampled before similarity between terms can be measured. It also means that additional context information can be added at any time without invalidating the information already produced. This is not feasible with most other word-space models. The approach used by Grefenstette (1994) and Curran (2004) requires the re-computation of all non-linear weights if new data is added, although some of these weights can be approximated when adding new data incrementally. Similarly, new data can be *folded* into a reduced LSA space, but there is no guarantee that the original smoothing will apply correctly to the new data (Sahlgren, 2005).

3 Weights

Our initial experiments using Random Indexing to extract synonymy relations produced worse results than those using full vector measures, such as JACCARD (Curran, 2004), when the full vector is weighted. We experiment using weight functions with Random Indexing.

Only a linear weighting scheme can be applied while maintaining incremental sampling. While incremental sampling is part of the rationale behind its development, it is not required for Random Indexing to work as a dimensionality reduction technique.

To this end, we revise Random Indexing to enable us to use weight functions. For each unique

IDENTITY	1.0	FREQ	$f(w,r,w')$
RELFREQ	$\frac{f(w,r,w')}{f(w,*,*)}$	TF-IDF	$\frac{f(w,r,w')}{n(*,r,w')}$
TF-IDF†	$\frac{\log_2(f(w,r,w')+1)}{\log_2(1+\frac{N_{(r,w')}}{n(*,r,w')})}$	MI	$\log(\frac{p(w,r,w')}{p(w,*,*)p(*,r,w')})$
TTEST	$\frac{p(w,r,w')-p(*,r,w')p(w,*,*)}{\sqrt{p(*,r,w')p(w,*,*)}}$	GREF94	$\frac{\log_2(f(w,r,w')+1)}{\log_2(n(*,r,w')+1)}$
LIN98A	$\log(\frac{f(w,r,w')f(*,r*)}{f(*,r,w')f(w,r,*)})$	LIN98B	$-\log(\frac{n(*,r,w')}{N_w})$
CHI2	*cf. Manning and Schütze (1999)*	LR	*cf. Manning and Schütze (1999)*
DICE	$\frac{2p(w,r,w')}{p(w,*,*)+p(*,r,w')}$		

Table 1: Weight Functions Evaluated

context attribute, a d length index vector will be generated. The context vector of a term w is then created by the weighted sum of each of its attributes. The results of the original Random Indexing algorithm are reproduced using frequency weighting (FREQ).

Weights are generated using the frequency distribution of each term and its contexts. This increases the overhead, as we must store the context attributes for each term. Rather than the context vector being generated by adding each individual context, it is generated by adding each the index vector for each unique context multiplied by its weight.

The time to calculate the weight of all attributes of all terms is negligible. The original technique scales to $O(dnm)$ in construction, for n terms and m unique attributes. Our new technique scales to $O(d(a + nm))$ for a non-zero context attributes per term, which since $a \ll m$ is also $O(dnm)$.

Following the notation of Curran (2004), a *context relation* is defined as a tuple (w, r, w') where w is a term, which occurs in some grammatical relation r with another word w' in some sentence. We refer to the tuple (r, w') as an *attribute* of w. For example, (dog, direct-obj, walk) indicates that dog was the direct object of walk in a sentence.

An asterisk indicates the set of all existing values of that component in the tuple.

$$(w, *, *) \equiv \{(r, w')|\exists(w, r, w')\}$$

The frequency of a tuple, that is the number of times a word appears in a context is $f(w, r, w')$. $f(w, *, *)$ is the *instance* or *token* frequency of the contexts in which w appears. $n(w, *, *)$ is the *type* frequency. This is the number of attributes of w.

$$\begin{aligned} f(w,*,*) &\equiv \textstyle\sum_{(r,w')\in(w,*,*)} f(w,r,w') \\ p(w,*,*) &\equiv \frac{f(w,*,*)}{f(*,*,*)} \\ n(w,*,*) &\equiv |(w,*,*)| \\ N_w &\equiv |\{w|n(w,*,*) > 0\}| \end{aligned}$$

Most experiments limited weights to the positive range; those evaluated with an unrestricted range are marked with a $^{\pm}$ suffix. Some weights were also evaluated with an extra $\log_2(f(w, r, w') + 1)$ factor to promote the influence of higher frequency attributes, indicated by a LOG suffix. Alternative functions are marked with a dagger.

The context vector of each term w is thus:

$$\bar{w} = \sum_{(r,w')\in(w,*,*)} (r, \vec{w'})\,\mathrm{wgt}(w, r, w')$$

where $(r, \vec{w'})$ is the index vector of the context (r, w'). The weights functions we evaluate are those from Curran (2004) and are given in Table 1.

4 Semantic Similarity

The first use of Random Indexing was to measure semantic similarity using distributional similarity. Kanerva et al. (2000) used Random Indexing to find the best synonym match in Test of English as a Foreign Language (TOEFL). TOEFL was used by Landauer and Dumais (1997), who reported an accuracy 36% using un-normalised vectors, which was improved to 64% using LSA. Kanerva et al. (2000) produced an accuracy of 48–51% using the same type of document based contexts and Random Indexing, which improved to 62–70% using narrow context windows. Karlgren and Sahlgren (2001) improved this to 72% using lemmatisation and POS tagging.

4.1 Distributional Similarity

Measuring distributional similarity first requires the extraction of context information for each of the vocabulary terms from raw text. The contexts for each term are collected together and counted, producing a vector of context attributes and their frequencies in the corpus. These terms are then compared for similarity using a nearest-neighbour search based on distance calculations between the statistical descriptions of their contexts.

The simplest algorithm for finding synonyms is a k-nearest-neighbour search, which involves pairwise vector comparison of the context vector of the target term with the context vector of every other term in the vocabulary.

We use two types of context extraction to produce both high and low quality context descriptions. The high quality contexts were extracted from grammatical relations extracted using the SEXTANT relation extractor (Grefenstette, 1994) and are lemmatised. This is the same data used in Curran (2004).

The low quality contexts were extracted taking a window of one word to the left and right of the target term. The context is marked as to whether it preceded or followed the term. Curran (2004) found this extraction technique to provided reasonable results on the non-speech portion of the BNC when the data was lemmatised. We do not lemmatise, which produces noisier data.

4.2 Bilingual Lexicon Acquisition

A variation on the extraction of synonymy relations, is the extraction of bilingual lexicons. This is the task of finding for a word in one language words of a similar meaning in a second language. The results of this can be used to aid manual construction of resources or directly aid translation.

This task was first approached as a distributional similarity-like problem by Brown et al. (1988). Their approach uses aligned corpora in two or more languages: the *source language*, from which we are translating, and the *target language*, to which we are translating. For a each aligned segment, they measure *co-occurrence scores* between each word in the source segment and each word in the target segment. These co-occurrence scores are used to measure the similarity between source and target language terms

Sahlgren and Karlgren's approach models the problem as a distributional similarity problem using the paragraph as context. In Table 2, the source language is limited to the words a, b and c and the target language to the words x, y and z. Three paragraphs in each of these languages are presented as pairs of translations labelled as a context: aaabbc is translated as xxyzzz and labelled context I. The frequency weighted context vector for a is {I:3, III:2} and for x is {I:2, II:1, III:1}.

Source Language	Context	Target Language
aaabbc	I	xxyzzz
bcc	II	wxy
aab	III	xzz

Table 2: Paragraph Aligned Corpora

A translation candidate for a term in the source language is found by measuring the similarity between its context vector and the context vectors of each of the terms in the target language. The most similar target language term is the most likely translation candidate.

Sahlgren and Karlgren (2005) use Random Indexing to produce the context vectors for the source and target languages. We re-implement their system and apply weighting functions in an attempt to achieve improved results.

5 Experiments

For the experiments extracting synonymy relations, high quality contexts were extracted from the non-speech portion of the British National Corpus (BNC) as described above. This represents 90% of the BNC, or 90 million words.

Comparisons between low frequency terms are less accurate than between high frequency terms as there is less evidence describing them (Curran and Moens, 2002). This is compounded in randomised vector techniques because the randomised nature of the representation means that a low frequency term may have a similar context vector to a high frequency term while not sharing many contexts. A frequency cut-off of 100 was found to balance this inaccuracy with the reduction in vocabulary size. This reduces the original 246,046 word vocabulary to 14,862 words. Experiments showed $d = 1000$ and $\epsilon = 10$ to provide a balance between speed and accuracy.

Low quality contexts were extracted from portions of the entire of the BNC. These formed corpora of 100,000, 500,000, 1 million, 5 million, 10 million, 50 million and 100 million words, chosen from random documents. This allowed us test the effect of both corpus size and context quality. This produced vocabularies of between 10,380 and 522,163 words in size. Because of the size of the smallest corpora meant that a high cutoff would remove to many terms for a fair test, a cutoff of 5 was applied. The values $d = 1000$ and $\epsilon = 6$ were used.

For our experiments in bilingual lexicon acquisition we follow Sahlgren and Karlgren (2005). We use the Spanish-Swedish and the English-German portions of the Europarl corpora (Koehn, 2005).[1] These consist of 37,379 aligned paragraphs in Spanish–Swedish and 45,556 in English-German. The text was lemmatised using Connexor Machinese (Tapanainen and Jävinen, 1997)[2] producing vocabularies of 42,671 terms of Spanish, 100,891 terms of Swedish, 40,181 terms of English and 70,384 terms of German. We use $d = 600$ and $\epsilon = 6$ and apply a frequency cutoff of 100.

6 Evaluation Measures

The simplest method for evaluation is the direct comparison of extracted synonyms with a manually created gold standard (Grefenstette, 1994). To reduce the problem of limited coverage, our evaluation of the extraction of synonyms combines three electronic thesauri: the Macquarie, Roget's and Moby thesauri.

We follow Curran (2004) and use two performance measures: direct matches (Direct) and inverse rank (InvR). Direct is the number of returned synonyms found in the gold standard. InvR is the sum of the inverse rank of each matching synonym, e.g. matches at ranks 3, 5 and 28 give an inverse rank score of $\frac{1}{3} + \frac{1}{5} + \frac{1}{28}$. With at most 100 matching synonyms, the maximum InvR is 5.187. This more fine grained as it incorporates the both the number of matches and their ranking.

The same 300 single word nouns were used for evaluation as used by Curran (2004) for his large scale evaluation. These were chosen randomly from WordNet such that they covered a range over the following properties: *frequency*, *number of senses*, *specificity* and *concreteness*. On average each evaluation term had 301 gold-standard synonyms. For each of these terms, the closest 100 terms and their similarity scores were extracted.

Weight	Direct	InvR
Freq	**2.87**	**0.94**
Identity	3.18	0.95
RelFreq	2.87	0.94
tf-idf	0.30	0.07
tf-idf†	**3.92**	**1.39**
MI	1.52	0.54
MILog	3.38	1.39
MI$^{\pm}$	1.87	0.65
MILog$^{\pm}$	**3.49**	**1.41**
TTest	1.06	0.52
TTestLog	1.53	0.62
TTest$^{\pm}$	1.06	0.52
TTestLog$^{\pm}$	1.52	0.61
Gref94	2.82	0.86
Lin98a	1.52	0.50
Lin98b	2.95	0.84
Chi2	0.46	0.25
Dice	3.32	1.11
DiceLog	2.56	0.81
LR	1.96	0.58

Table 3: Evaluation of synonym extraction

For the evaluation of bilingual lexicon acquisition we use two online lexical resources used by Sahlgren and Karlgren (2005) as gold standards: Lexin's online Swedish-Spanish lexicon[3] and TU Chemnitz' online English-German dictionary.[4] Each of the elements in a compound or multi-word expression is treated as a potential translation. The German abblendlicht (low beam light) is treated as a translation candidate for low, beam and light separately.

Low coverage is more of problem than in our thesaurus task as we have not used combined resources. There are an average of 19 translations for each of the 3,403 Spanish terms and 197 translations for each of the 4,468 English terms. The English-German translation count is skewed by the presence of connectives in multi-word expressions, such as of and on, producing mistranslations. Sahlgren and Karlgren (2005) provide good commentary on the evaluation of this task.

Spanish and English are used as the source languages. The 200 closest terms in the target language are found for all terms in both the source vocabulary and the gold-standards.

We measure the Direct score and InvR as above. In addition we measure the precision of the closest translation candidate, as used in Sahlgren and Karlgren (2005).

[1] http://www.statmt.org/europarl/

[2] http://www.connexor.com/

[3] http://lexin.nada.kth.se/sve-spa.shtml

[4] http://dict.tu-chemnitz.de/

Weight	English-German			Spanish-Swedish		
	DIRECT	Precision	INVR	DIRECT	Precision	INVR
FREQ	6.1	58%	0.97	0.8	47%	0.53
IDENTITY	6.0	58%	0.91	0.8	47%	0.53
RELFREQ	6.1	58%	0.97	0.8	47%	0.53
TF-IDF	4.9	53%	0.84	0.8	43%	0.50
TF-IDF†	6.3	58%	0.94	0.8	47%	0.53
MI	2.3	58%	0.76	0.8	48%	0.56
MILOG	2.1	58%	0.76	0.8	49%	0.56
MI$^{\pm}$	4.6	57%	0.86	0.8	46%	0.53
MILOG$^{\pm}$	4.6	57%	0.87	0.8	47%	0.54
TTEST	2.1	57%	0.75	0.8	48%	0.56
TTESTLOG	1.9	56%	0.72	0.8	46%	0.54
TTEST$^{\pm}$	4.3	57%	0.85	0.8	45%	0.53
TTESTLOG$^{\pm}$	4.0	56%	0.80	0.8	46%	0.53
GREF94	6.1	58%	0.95	0.8	48%	0.54
LIN98A	4.0	59%	0.82	0.8	48%	0.56
LIN98B	5.9	58%	0.91	0.8	48%	0.54
CHI2	3.1	50%	0.71	0.7	41%	0.48
DICE	5.7	58%	0.95	0.8	47%	0.53
DICELOG	4.7	57%	0.90	0.8	46%	0.52
LR	4.5	57%	0.86	0.8	47%	0.54

Table 4: Evaluation of bilingual lexicon extraction

7 Results

Table 3 shows the results for the experiments extracting synonymy. The basic Random Indexing algorithm (FREQ) produces a DIRECT score of 2.87, and an INVR of 0.94. It is interesting that the only other linear weight, IDENTITY, produces more accurate results. This shows high frequency, low information contexts reduce the accuracy of Random Indexing. IDENTITY removes this effect by ignoring frequency, but does not address the information aspect. A more accurate weight will consider the information provided by a context in its weighting.

There was a large variance in the effectiveness of the other weights and most proved to be detrimental to Random Indexing. TF-IDF was the worst, reducing the DIRECT score to 0.30 and the INVR to 0.07. TF-IDF†, which is a log-weighted alternative to TF-IDF, produced very good results.

With the exception of DICELOG, adding an additional log factor improved performance (TF-IDF†, MILOG and TTESTLOG). Unrestricted ranges improved the MI family, but made no difference to TTEST. Grefenstette's variation on TF-IDF (GREF94) does not perform as well as TF-IDF†, and Lin's variations on MI$^{\pm}$ (LIN98A, LIN98B) do not perform as well as MILOG$^{\pm}$.

MILOG$^{\pm}$ had a higher INVR than TF-IDF†, but a lower DIRECT score, indicating that it forces more correct results to the top of the results list, but also forces some correct results further down so that they no longer appear in the top 100.

Weight	BNC		LARGE	
	DIRECT	INVR	DIRECT	INVR
FREQ	8.9	0.93	7.2	0.85
TF-IDF†	11.8	1.39	12.5	1.50
MILOG$^{\pm}$	10.5	1.41	13.8	1.75

Table 5: Evaluation of Random Indexing using a very large corpus

The effect of high frequency contexts is increased further as we increase the size of the corpus. Table 5 presents results using the 2 billion word corpus used by Curran (2004). This consists of the non-speech portion of the BNC, the Reuter's Corpus Volume 1 and most of the English news holdings of the LDC in 2003. Contexts were extracted as presented in Section 4. A frequency cutoff of 100 was applied and the values $d = 1000$ and $\epsilon = 5$ for FREQ and $\epsilon = 10$ for the improved weights were used.

We see that the very large corpus has reduced the accuracy of frequency weighted Random Indexing. In contrast, our two top performers have both substantially increased in accuracy, presenting a 75–100% improvment in performance over FREQ. MILOG$^{\pm}$ is more accurate than TF-IDF† for both measures of accuracy now, indicating it is a better weight function for very large data sets.

7.1 Bilingual Lexicon Acquisition

When the same function were applied to the bilingual lexicon acquisition task we see substantially different results: neither the improvement nor the extremely poor results are found (Table 4).

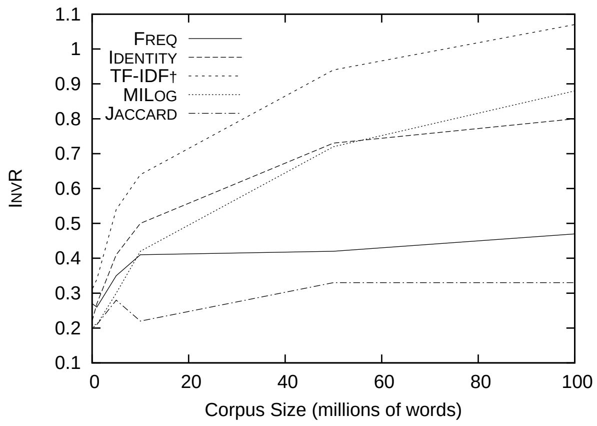

Figure 1: Random Indexing using window-based context

In the English-German corpora we replicate Sahlgren and Karlgren's (2005) results, with a precision of 58%. This has a DIRECT score of 6.1 and an INVR of 0.97. The only weight to make an improvement is TF-IDF†, which has a DIRECT score of 6.3, but a lower INVR and all weights perform worse in at least one measure.

Our results for the Spanish-Swedish corpora show similar results. Our accuracy is down from that in Sahlgren and Karlgren (2005). This is explained by our application of the frequency cut-off to both the source and target languages. There are more weights with higher accuracies, and fewer with significantly lower accuracies.

7.2 Smaller Corpora

The absence of a substantial improvement in bilingual lexicon acquisition requires further investigation. Three main factors differ between our monolingual and bilingual experiments: that we are smoothing a homogeneous data set in our monolingual experiments and a heterogeneous data set in our bilingual experiments; we are using local grammatical contexts in our monolingual experiments and paragraph contexts in our bilingual experiments; and, the volume of raw data used in our monolingual experiments is many times that used in our bilingual experiments.

Figure 1 presents results for corpora extracted from the BNC using the window-based context. Results are shown for the original Random Indexing (FREQ) and using IDENTITY, MILOG$^{\pm}$ and TF-IDF†, as well as for the full vector measurement using JACCARD measure and the TTEST$^{\pm}$ weight (Curran, 2004). Of the Random Indexing results FREQ produces the lowest overall results. It performs better than MILOG$^{\pm}$ for very small corpora, but produces near constant results for greater corpus sizes. Curran and Moens (2002) found that increasing the volume of input data increased the accuracy of results generated using a full vector space model. Without weighting, Random Indexing fails this, but after weighting is applied Curran and Moens' results are confirmed.

The quality of context extracted influences how weights perform individually, but Random Indexing using weights still outperforms not using weights. The relative performance of MILOG$^{\pm}$ has been reduced when compared with TF-IDF†, but is still greater then FREQ.

Gorman and Curran (2006) showed Random Indexing to be much faster than full vector space techniques, but with a 46–56% reduction in accuracy compared to using JACCARD and TTEST$^{\pm}$. Using the MI$^{\pm}$ weight kept the improvement in speed but with only a 10–18% reduction in accuracy. When JACCARD and TTEST$^{\pm}$ are used with our low quality contexts they perform consistently worse that Random Indexing. This indicates Random Indexing is stable in the presence of noisy data. It would be interesting to further compare these results to those produced by LSA.

The results we have presented have shown that applying weights to Random Indexing can improve its performance for thesaurus extraction tasks. This improvement is dependent on the volume of raw data used to generate the context information. It is less dependent on the quality of contexts extracted.

What we have not shown is whether this extends to the extraction of bilingual lexicons. The bilingual corpora have 12-16 million words per language, and for this sized corpora we already see substantial improvement with corpora as small as 5 million words (Figure 1). It may be that extracting paragraph-level contexts is not well suited to weighting, or that the heterogeneous nature of the aligned corpora reduces the meaningfulness of weighting. There is also the question as to whether it can be applied to all languages. There is a lack of freely available large-scale multi-lingual resources that makes this difficult to examine.

8 Conclusion

We have applied weighting functions to the vector space approximation Random Indexing. For large data sets we found a significant improvement when weights were applied. For smaller data sets we found that Random Indexing was sufficiently robust that weighting had at most a minor effect.

Our weighting schemes removed the possibility of incremental learning of the term space. An interesting direction would be the development of algorithms that allowed the incremental application of weights, perhaps by re-weighting vectors when a new context is learned.

Other areas left open for investigation are the interaction between Random Indexing, weights and the type of context extracted, the use of large-scale bilingual corpora, the acquisition of lexicons for non-Indo-European languages and across language family boundaries, and the difference in effect term and paragraph/document contexts for thesaurus extraction.

We have demonstrated that the accuracy of Random Indexing can be improved by applying weight functions, increasing accuracy by up to 50% on the BNC and 100% on a 2 billion word corpus.

Acknowledgements

We would like to thank Magnus Sahlgren for generously supplying his training and evaluation data and our reviewers for their helpful feedback and corrections. This work has been supported by the Australian Research Council under Discovery Project DP0453131.

References

Dimitris Achlioptas. 2001. Database-friendly random projections. In *Symposium on Principles of Database Systems*, pages 274–281, Santa Barbara, CA, USA, 21–23 May.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, Robert L. Mercer, and Paul S. Roossin. 1988. A statistical approach to language translation. In *Proceedings of the 12th Conference on Computational linguistics*, pages 71–76, Budapest, Hungry, 22–27 August.

James R. Curran and Marc Moens. 2002. Scaling context space. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 231–238, Philadelphia, PA, USA, 7–12 July.

James R. Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, University of Edinburgh.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. The MIT Press, Cambridge, MA, USA.

James Gorman and James R. Curran. 2006. Scaling distributional similarity to large corpora. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, 17–21 July. To appear.

Gregory Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Boston.

Robert Hecht-Nielsen. 1994. Context vectors: general purpose approximate meaning representations self-organized from raw data. *Computational Intelligence: Imitating Life*, pages 43–56.

William B. Johnson and Joram Lindenstrauss. 1984. Extensions to Lipshitz mapping into Hilbert space. *Contemporary mathematics*, 26:189–206.

Pentti Kanerva, Jan Kristoferson, and Anders Holst. 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, page 1036, Philadelphia, PA, USA, 13–15 August.

Jussi Karlgren and Magnus Sahlgren. 2001. From words to understanding. In Y. Uesaka, P. Kanerva, and H Asoh, editors, *Foundations of Real-World Intelligence*, pages 294–308. CSLI Publications, Stanford, CA, USA.

Samuel Kaski. 1998. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of the International Joint Conference on Neural Networks*, pages 413–418. Piscataway, NJ, USA, 31 July–4 August.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit X*, Phuket, Thailand, 12–16 September.

Thomas K. Landauer and Susan T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240, April.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA, USA.

Christos H. Papadimitriou, Hisao Tamaki, Prabhakar Raghavan, and Santosh Vempala. 1998. Latent semantic indexing: A probabilistic analysis. In *Proceedings of the 17th ACM Symposium on the Principle of Database Systems*, pages 159–168, Seattle, WA, USA, 2–4 June.

Magnus Sahlgren and Jussi Karlgren. 2005. Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Journal of Natural Language Engineering, Special Issue on Parallel Texts*, 11(3), June.

Magnus Sahlgren. 2005. An introduction to random indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering*, Copenhagen, Denmark, 16 August.

Pasi Tapanainen and Timo Jävinen. 1997. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 64–71, 31 March–3 April.