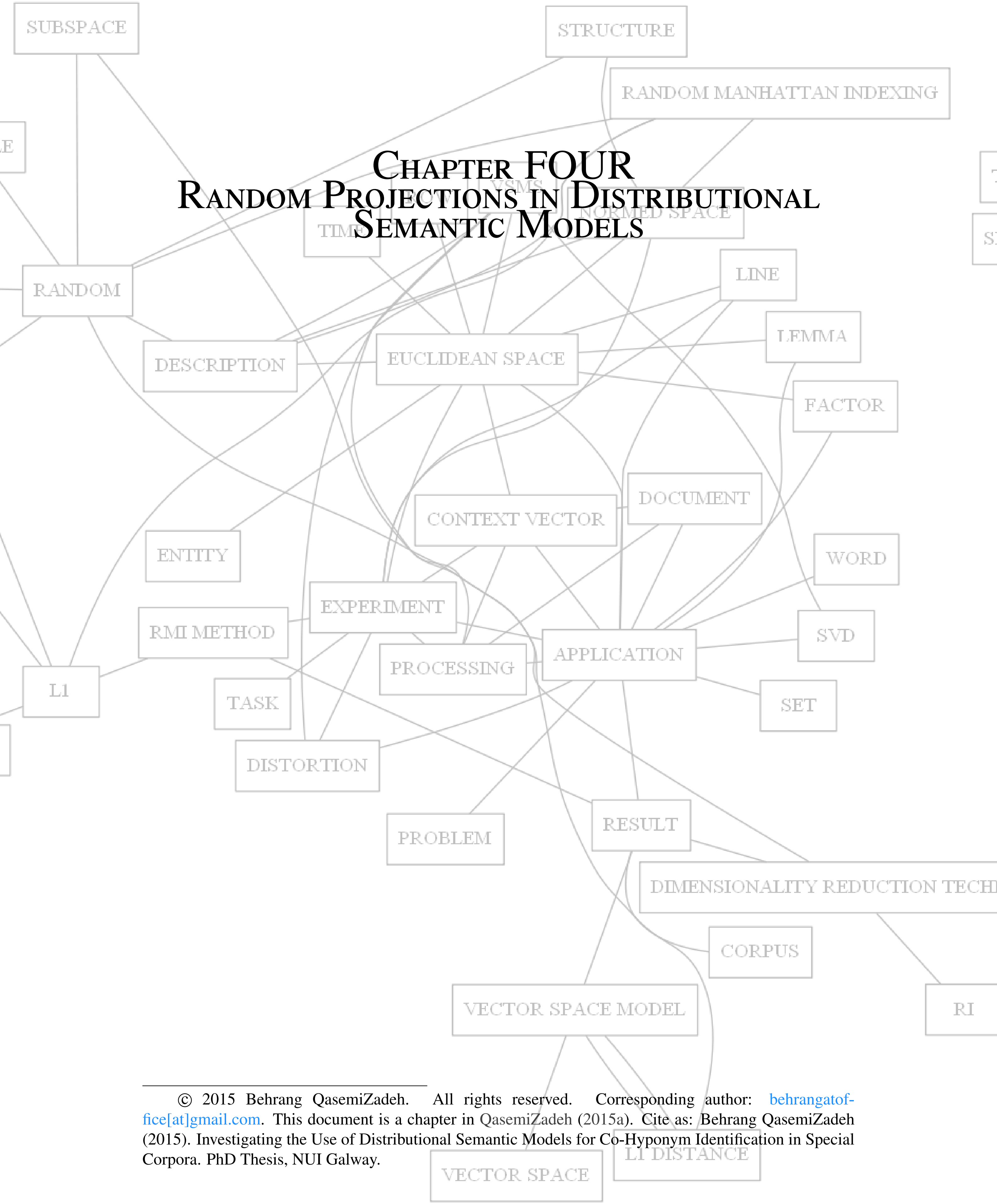
Chapter FOUR
Random Projections in Distributional Semantic Models



Corresponding author: behrangatoffice[at]gmail.com. This document is a chapter in QasemiZadeh (2015a). Cite as: Behrang QasemiZadeh (2015). Investigating the Use of Distributional Semantic Models for Co-Hyponym Identification in Special Corpora. PhD Thesis, NUI Galway.

This page is intentionally left blank.

Contents

This page is intentionally left blank.

List of Figures

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

Chapter 4

Random Projections in Distributional Semantic Models

Random projections are mathematical tools that have been widely used in algorithm design. They have had a number of significant contributions in several domains, such as the applications of machine learning techniques to big data. At the expense of negligible loss in the accuracy of the estimated distances between vectors, these methods reduce the size of vectors to enhance the performance of processes. In distributional semantic models, random indexing is one of the widely-used methods that can be understood using the random projections theorems. In this chapter, the principles of random projections are employed in order to reintroduce random indexing and propose new dimensionality reduction methods for the ℓ_1-normed spaces.

This chapter starts with recapping the *curse of dimensionality* problem in distributional semantic models and enumerating a number of motivations for the proposed methods in Section 4.1. In Section 4.2, the random indexing technique is explained and justified mathematically. In Section 4.3, by extending the use of random projections to ℓ_1-normed spaces, a novel technique called random Manhattan indexing (RMI) is introduced. In Section 4.4, RMI and RI are compared, followed by a summary in Seciton 4.5.[1]

[1] Section 4.2 is mainly based on QasemiZadeh (2015b) and QasemiZadeh and Handschuh (2015). Section 4.3.1 and 4.3.2 are based on Zadeh and Handschuh (2014a) and Zadeh and Handschuh (2014b), respectively.

4.1 Introduction

In order to model any aspect of the meanings in language, distributional semantic models exploit patterns of co-occurrences. These methods tie the usage context of linguistic entities (e.g., words and phrases) to their meaning. Hence, meanings are assessed by quantification of the distributional similarities of linguistic entities. An intuitive, mathematically well-defined model to represent and process such distributional similarities—amongst other representation frameworks—is vector space.

Recall from Chapter 2, particularly Section 2.2.1, in a vector space model, each element $\vec{s_i}$ of the standard basis (i.e., informally each dimension of the vector space) represents a context element. Given n context elements, a linguistic entity whose meaning is being analysed is expressed by a vector $\vec{v}$ as a linear combination of $\vec{s_i}$ and scalars $\alpha_i \in \mathbb{R}$ such that $\vec{v} = \alpha_1 \vec{s_1} + \cdots + \alpha_n \vec{s_n}$. The value of α_i is acquired from the frequency of the co-occurrences of the entity that $\vec{v}$ represents and the context element that $\vec{s_i}$ represents. As a result, the values assigned to the coordinates of a vector, that is, α_i, exhibit the correlation of an entity and the context elements in an n-dimensional real vector space $\mathbb{R}^n$.

In this vector space, similarities of vectors are understood to indicate similarities of the meanings of linguistic entities that they represent. In order to assess the similarity between vectors, a vector space V is endowed with a *norm* structure.[1] A norm $||.||$ is a function that maps vectors from V to the set of non-negative real numbers, that is, $V \mapsto [0, \infty)$. The pair of $(V, ||.||)$ is then called a *normed* space. In a normed space, the similarity between vectors is assessed by their distances. The distance between vectors is defined by a function that satisfies certain axioms and assigns a real value to each pair of vectors, that is,

$$dist : V \times V \mapsto \mathbb{R}, \quad d(\vec{v}, \vec{t}) = ||\vec{v} - \vec{u}||. \tag{4.1}$$

The smaller the distance between two vectors, the more similar they are.

Amongst several choices, an ℓ_2-normed-based metric—such as the Euclidean distance and the cosine similarity—is an innate choice.

Euclidean space is the most familiar example of a normed space. It is a vector space that is endowed by the ℓ_2 norm. In Euclidean space, the ℓ_2 norm—which is also called the Euclidean norm—of a vector $\vec{v} = (v_1, \cdots, v_n)$ is defined as:

$$||\vec{v}||_2 = \sqrt{\sum_{i=1}^{n} v_i^2}. \tag{4.2}$$

Using the definition of distance given in Equation 4.1 and the ℓ_2 norm, the Euclidean distance is measured as:

$$dist_2(\vec{v}, \vec{u}) = ||\vec{v} - \vec{u}||_2 = \sqrt{\sum_{i=1}^{n} (v_i - u_i)^2}. \tag{4.3}$$

In ℓ_2-normed vector spaces, various similarity metrics are defined using different normalisation of the Euclidean distance between vectors, for example, the *cosine similarity*.

[1] Please note other structures than norm can be employed to assess the similarities.

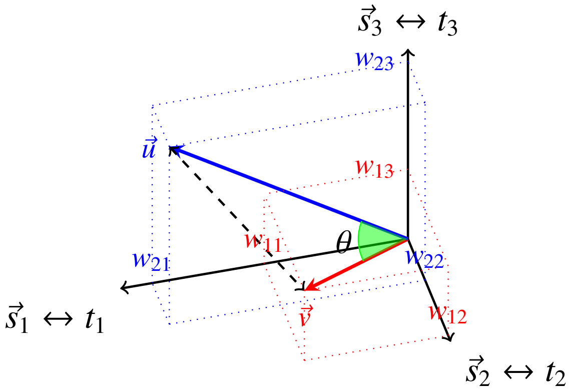

$\vec{s}_3 \leftrightarrow t_3$
w_{23}
$\vec{u}$
w_{13}
θ
w_{11}
w_{21}
w_{22}
$\vec{s}_1 \leftrightarrow t_1$
$\vec{v}$
w_{12}
$\vec{s}_2 \leftrightarrow t_2$

Figure 4.1: Illustration of a *document-by-term* model consisting of 2 documents and 3 terms. Each element of the standard basis s_i (i.e., each dimension), represents one of the 3 terms in the model. The 3-dimensional vectors $\vec{v} = (w_{11}, w_{12}, w_{13})$ and $\vec{u} = (w_{21}, w_{22}, w_{23})$ represent the two documents in the model. The dashed line shows the Euclidean distance between the vectors. Similarly, the cosine of the angel between the vectors, $\cos(\theta)$, defines the cosine similarity between them.

A classic Salton et al.'s (1975) document-by-term model is, perhaps, the most familiar example of the above-described vector space model (VSM). Given n distinct terms t and a number of documents d, each document d_i is represented by an n-dimensional vector $\vec{d_i} = (w_{i1}, \cdots, w_{in})$, where w_{ij} is a numeric value that associates the document d_i to the term t_j, for $1 < j < n$. For instance, w_{ij} may correspond to the frequency of the terms t_j in the document d_i. For a collection of m documents, a *document-by-term* matrix $\mathbf{M}_{m \times n}$ denotes the constructed vector space. Using the *bag of words* hypothesis, it is assumed that the relevance of documents can be assessed by counting terms that appear in the documents, independent of their order or syntactic usage patterns. Documents with similar vectors are thus assumed to share the same meaning. Using the ℓ_2-norm, the similarity between documents is then calculated by the Euclidean distance or the cosine similarity shown in Figure 4.1.

As discussed in Chapter 2, when the number of entities in a VSM increases, the number of context elements employed for capturing similarities between them surges. As a result, usually high-dimensional vectors, in which most elements are zero, represent entities. However, when the dimension of vectors in a VSM increases, the discriminatory power of the VSM diminishes. This results in setbacks known as the *curse of dimensionality*. Hence, the curse of dimensionality is tackled using a *dimensionality reduction* technique.

Dimensionality reduction can be achieved using a number of methods as an auxiliary process that is followed by the construction of a VSM—ranging from heuristic-based selection process to ad hoc matrix factorisation techniques such as singular value decomposition (see Section 2.3.3). The use of these dimensionality reduction techniques, however, is hampered by a number of factors.

Firstly, a VSM at the original high dimension must be first constructed. Following the construction of the VSM, the dimension of the VSM is reduced in an independent process. The VSM with the reduced dimensionality is thus available for processing only after the whole sequence of these processes. However, construction of the VSM at its original dimension is computationally expensive (e.g., all the co-occurrences must be collected and stored) and the delayed access to the VSM with the reduced dimensionality is not desirable.

Secondly, reducing the dimension of vectors using the methods listed above is of high computational complexity. For instance, mapping $\mathbb{R}^n$ onto $\mathbb{R}^m$ using SVD truncation demands a process of the time complexity $\mathrm{O}(n^2m)$ and space complexity $\mathrm{O}(n^2)$.[1] Similarly, in a heuristic-based selection process, the collected frequencies for each of the context elements must be assessed. Depending on the employed heuristic, this process can be resource-intensive, too; for example, the collected frequencies are often required to be sorted by some criteria.

Last but not least, these methods are *data-sensitive*: if the structure of the data being analysed changes—that is, if either linguistic entities or context elements are updated, for example, some are removed or new ones are added—the dimensionality reduction process is required to be repeated and reapplied to the whole VSM in order to reflect these updates. The use of feature selection techniques or truncated SVD, therefore, may not be desirable in several applications, particularly when dealing with frequently updated big text-data.

Random projections are mathematical tools that are employed to implement alternative dimensionality reduction techniques that can alleviate the aforementioned problems. Random projections map high-dimensional vector spaces onto a low-dimension subspace using matrices consisting of randomly generated vectors that guarantee the preservation of distances between vectors. Hence, random projections are used to design dimensionality reduction techniques that (a) bypass a number of computations in the classic dimensionality reduction techniques (e.g., the computation of orthogonal subspaces or selecting context elements), and (b) merge the dimensionality reduction process into the process of vector space construction to suggest an incremental—thus scalable—technique for the construction of VSMs directly at a reduced dimensionality.

In the context of distributional semantic models, the widely-employed random indexing technique can be justified using the mathematical principles of random projections. Random indexing (RI) is an incremental method for the construction of vector spaces at a reduced dimensionality. It was first introduced by Kanerva et al. (2000) and further propounded by Sahlgren (2005). Sahlgren (2005) delineates the RI method as a two-step procedure that consists of the construction of (a) *index vectors* and (b) *context vectors*.

In the first step, each context element is assigned *exactly* to one *index vector*. Sahlgren (2005) indicates that index vectors are high-dimensional, randomly generated vectors, in which most of the elements are set to 0 and only *a few* to 1 and -1. In the second step, during the construction of *context vectors*, each target entity is assigned to a *zero vector* (i.e., all the elements of the vector are zero) that has the same dimension as the index vectors. For each occurrence of an entity, which is represented by $\vec{v}_{e_i}$, and a context element, which is represented by $\vec{r}_{c_k}$, the context vector for the entity is accumulated by

[1] It is worth mentioning that the use of incremental techniques can relax these requirements to an extent (e.g., see Brand, 2006).

the index vector of the context element, that is, $\vec{v}_{e_i} = \vec{v}_{e_i} + \vec{r}_{c_k}$. The result is a vector space model constructed directly at reduced dimension.

Both Sahlgren (2005) and Kanerva et al. (2000) introduce the random indexing method in a mathematical framework other than random projections—that is the sparse distributed memory (SDM).[1] The random indexing method was then developed and justified by Kanerva et al. (2000) as one of the extensions of SDM, without providing a mathematical justification for the suggested two-step procedure and the method's parameters—that is, the dimension of index vectors and the proportion of zero and non-zero elements in them.

In the remainder of this chapter, the random indexing technique is revisited and explained using theorems of random projections, which are refined by advances in statistics. In contrast to the previous delineations of this method, the provided description gives an understanding of the method which can be used for setting the method's parameters, recognising the limits of its use, and extending it to normed spaces other than ℓ_2.

In Section 4.2, random projections in Euclidean spaces—hence random indexing—is refined using mathematical theorems, which are verified by empirical experiments. Accordingly, Section 4.3 describes random projections in ℓ_1-normed spaces, and introduce the random Manhattan indexing technique—that is, a method similar to RI but for estimating city block distances. The differences between RI and RMI are reviewed in Section 4.4. Finally, this chapter concludes with a discussion and summary in Section 4.5.

4.2 Random Projections in Euclidean Spaces

In Euclidean spaces, random projections are elucidated by Johnson and Lindenstrauss's (1984) lemma (JL lemma). Given an ϵ, $0 < \epsilon < 1$, the JL lemma states that for any set of p vectors in a high n-dimensional Euclidean space $\mathbb{E}^n$,[2] there exists a mapping onto an m-dimensional space $\mathbb{E}^m$, for $m \geq m_0 = O(^{\log p}/_{\epsilon^2})$, that does not distort the distances between any pair of vectors, with high probability, by a factor more than $1 \pm \epsilon$. This mapping can be expressed by

$$\mathbf{M}'_{p\times m} = \mathbf{M}_{p\times n}\mathbf{R}_{n\times m}, \quad m \ll p, n, \tag{4.4}$$

where $\mathbf{R}_{n\times m}$ is often called the random projection matrix, and $\mathbf{M}_{p\times n}$ and $\mathbf{M}'_{p\times m}$ denote the p vectors in $\mathbb{E}^n$ and $\mathbb{E}^m$, respectively. According to the JL lemma, if the distance between any pair of vectors $\vec{v}$ and $\vec{u}$ in $\mathbf{M}$ is given by the $dist_{\text{Euc}}(\vec{v}, \vec{u})$, and their distance in $\mathbf{M}'$ is given by $dist'_{\text{Euc}}(\vec{v}, \vec{u})$, then there exists an $\mathbf{R}$ such that

$$(1-\epsilon)dist'_{\text{Euc}}(\vec{v}, \vec{u}) \leq dist_{\text{Euc}}(\vec{v}, \vec{u}) \leq (1+\epsilon)dist'_{\text{Euc}}(\vec{v}, \vec{u}).^{3} \tag{4.5}$$

Instead of the original n-dimensional vector space and at the expense of negligible amount of error ϵ, the distance between $\vec{v}$ and $\vec{u}$ can be calculated in the m-dimensional vector space. Accordingly, since $m \ll n$, the time and the space complexity for the computation

[1] For a brief introduction to sparse distributed memory see Kanerva (1993)

[2] $\mathbb{E}^n$ is an n-dimensional real vector space $\mathbb{R}^n$ endowed by the ℓ_2 norm.

[3] In addition, the lemma states that this mapping can be found in randomised polynomial time.

of distances can be reduced significantly. The random projection matrix **R** is stored for later usages, such as adding new entities to the vector space.

The JL lemma does not specify the projection matrix **R**. Finding **R** that satisfy the JL lemma is therefore the most important design decision when using random projections. Originally, Johnson and Lindenstrauss (1984) proved the lemma using an orthogonal projection onto a random m-dimensional subspace of the original vector space. Subsequent studies simplified the original proof and suggested several choices of **R** that resulted in projection techniques with enhanced computational efficiency (e.g., see Dasgupta and Gupta, 2003, for references). It is proved that a mapping that satisfies the JL lemma can be obtained, with a *high probability*, using a random projection **R** whose entries are independent and identically distributed (i.i.d.) and have zero mean and constant variance.[1]

Recently, Achlioptas (2001) shows that a sparse **R** with an *asymptotic Gaussian distribution*, whose elements r_{ij} are defined as

$$r_{ij} = \sqrt{s}\begin{cases} -1 & \textit{with probability } \frac{1}{2s} \\ 0 & \textit{with probability } 1 - \frac{1}{s} \\ 1 & \textit{with probability } \frac{1}{2s} \end{cases}, \tag{4.6}$$

for $s \in \{1, 3\}$, results in a mapping that also satisfies the JL lemma.[2]

Subsequent research showed that **R** can be constructed from even sparser vectors than what is suggested in Achlioptas (2001) (e.g., see Li et al., 2006b; Matoušek, 2008). Specifically, Li et al. (2006b) has proved that in a mapping of an n-dimensional real vector space by a sparse **R**, the JL lemma holds as long as $s = \mathrm{O}(n)$, for example, $s = \sqrt{n}$ or even $s = {}^{n}/_{\log(n)}$.

Using a sparse **R** that is given by Equation 4.6 reduces the number of multiplication operations in Equation 4.4 by the factor $\frac{1}{s}$ and thus speeds up the mapping process—that is, the computation of $\mathbf{M}'$. The larger the value of s, the sparser the random vector is; hence, at the expense of insignificant loss in the accuracy of the estimated distances, it is expected that the succeeding processes will be faster. Moreover, the multiplication of the scaling factor $\sqrt{s}$ can be postponed until after the mapping, or when it is necessary. Floating-point arithmetic operations, therefore, can be avoided during the computation of the mapping, which consequently enhances the computational as well as the memory complexity. Nonetheless, to say that a sparse **R** requires less space for its storage.

Apart from the sparse mapping, another major benefit when computing $\mathbf{M}'$ is obtained using the linearity of matrix multiplication. Each vector $\vec{v}_{e_i}$ in the original n-dimensional space, that is, ith row of **M**, can be represented as a weighted sum of the basis vectors

$$\vec{v}_{e_i} = w_{i1}\vec{s}_{c_1} + w_{i2}\vec{s}_{c_2} + \cdots + w_{in}\vec{s}_{c_n}, \tag{4.7}$$

[1] For the simplicity of theoretical analysis, it is often assumed that entries of **R** have the standard Gaussian distribution—that is, for each m-dimensional random vector **r** in **R**, $\mathbf{r} \sim \mathcal{N}_m(0, 1)$. According to the central limit theorem, the probability distribution of i.i.d. variables that have finite variance approaches a Gaussian distribution.

[2] The mapping in Equation 4.6 guarantees that distances are preserved with a probability of at least $1 - p^{-\gamma}$, for some $\gamma > 0$ (see Achlioptas (2001), for proof and explanation.)

where w_{ij}, $i \leq p$ and $j \leq n$ are derived from the frequency of the co-occurrences of the entity and context element that $\vec{v}_{e_i}$ and $\vec{s}_{c_k}$ represent, respectively. By the basic properties of the matrix multiplication, the projection of $\vec{v}_{e_i}$ in $\mathbf{M}'$ is given by

$$\vec{v}'_{e_i} = \vec{v}_{e_i}\mathbf{R} = w_{i1}\vec{s}_{c_1}\mathbf{R} + w_{i2}\vec{s}_{c_2}\mathbf{R} + \cdots + w_{in}\vec{s}_{c_n}\mathbf{R}. \tag{4.8}$$

In turn, since, by definition, all the elements of the standard basis $\vec{s}_{c_k}$ are zero except the kth element, which is equal to 1, the statement given in Equation 4.8 can be equally written as

$$\vec{v}'_{e_i} = w_{i1}\vec{r}_1 + w_{i2}\vec{r}_2 + \cdots + w_{in}\vec{r}_n, \tag{4.9}$$

where $\vec{r}_j$ is the jth row of $\mathbf{R}$. Equation 4.9 means that row vectors $\vec{v}'_{e_i}$, thus $\mathbf{M}'$, can be computed directly without necessarily constructing the whole matrix $\mathbf{M}$. From one perspective, the jth row of $\mathbf{R}_{n\times m}$ represents a context element in the original vector space that is located at the jth column of $\mathbf{M}_{p\times n}$.[1] Therefore, a vector representation of an entity at a reduced dimension can be computed directly by accumulating the row vectors of $\mathbf{R}$ that represent the context elements that co-occur with the entity.

4.2.1 Improving the RI Algorithm: An Outcome of the Exposition

The RI technique can be reintroduced using the mathematical explanations given in the previous section. As can be understood, the RI technique can be seen as a dimensionality reduction technique for Euclidean spaces. RI implements a random projection that employs a random matrix $\mathbf{R}$ with an asymptotic Gaussian distribution (as it is expressed by Equation 4.4). The construction of index vectors—that is, the first step of RI—is equivalent to the construction of the random projection matrix $\mathbf{R}$, whose elements are given by Equation 4.6. Each index vector is a row of the random projection matrix $\mathbf{R}$. The second step of RI, the construction of context vectors, deals with the computation of $\mathbf{M}'$. Each context vector is a row of $\mathbf{M}'$, which is computed by the iterative process justified in Equation 4.9.

While in previous research the parameters of the RI method are left to be decided entirely through experiments (e.g., see Lupu, 2014; Polajnar and Clark, 2014), the adopted mathematical framework can be leveraged to provide a guideline for setting the RI's parameters. Using the JL lemma, a criterion for choosing the dimension of vector spaces constructed by the RI method at the reduced dimensionality (i.e., m in Equation 4.4) and the number of zero and non-zero elements in index vectors (i.e., s in Equation 4.6) are suggested.

In a VSM constructed using RI at a reduced dimensionality, the degree of preservation of distances between vectors in the original high dimension and at the reduced dimensionality m is determined by the number of vectors in the model and m. If the number of vectors (i.e., the number of entities that are modelled in the VSM) is fixed, then the larger m is, the better the Euclidean distances will be preserved at the reduced dimension m. In other words, the probability of preserving the pairwise distances increases as m increases.

[1] Informally, the jth dimension of the original n-dimensional vector space.

However, from the computational perspective, the lower the value of m is, the less computation is required for the construction of the VSM and the calculation of the distances, and therefore the better the efficiency is. From this perspective, the choice of dimensionality in RI-constructed VSMs is a trade-off between efficiency and accuracy. Similarly, the value of m can be seen as the capacity of a RI-constructed VSM for accommodating new entities. Therefore, compared to $m = 4000$ suggested in Kanerva et al. (2000) or $m = 1800$ in Sahlgren (2005), depending on the number of entities that are modelled in an experiment, m can be set to a smaller value such as $m = 400$.

The discussion above can be approached by investigating the distribution of the pairwise distances in the original high-dimensional vector space and the constructed vector space using RI (see also Stein, 2007). If the pairwise distances in the original space are and relatively small, then in order to be able to distinguish them, the distortion of the pairwise distances at the reduced dimensionality must be small (i.e., ϵ in Equation 4.5). If the number of entities in the model is fixed, then the distortion of the pairwise distances reduces when m increases. Hence, the distribution of the pairwise distances is a factor that can influence the chosen value for m.

Based on the results reported in Li et al. (2006b), when embedding an n-dimensional vector space onto a vector space of a reduced dimensionality m, the JL lemma holds—that is, pairwise Euclidean distances between vectors are preserved—as long as s in Equation 4.6 is $O(n)$. In text processing applications, the number of context elements and thus the dimension of vector spaces (i.e., n) is often very large. When using the random indexing method, therefore, even a careful choice such as $s = \sqrt{n}$ in Equation 4.6 results in very sparse random index vectors. In most text processing applications, therefore, by setting only 2 or 4 non-zero elements in index vectors, distances in the RI-constructed model resemble distances in the high n-dimensional model (for the mathematical proofs, see Li et al., 2006b, Appendix B).

It is worth reminding that if the dimension of index vectors (i.e., m) is fixed, then increasing the number of non-zero elements in index vectors causes additional distortions in the pairwise Euclidean distances. For index vectors of fixed dimensionality m, if the number of non-zero elements increases, then the probability of the orthogonality between index vectors decreases (see examples from a simulation in Figure 4.2). Hence, an increase in the number non-zero elements while m is fixed can stimulate distortions in pairwise distances. However, it is important to note that causing distortions in the pairwise distances can be beneficial; for example, it may reduce the effect of noise and foster assortment of similar context elements. As a result, distortions in the pairwise distances can be favourable in a number of applications.

To verify the theoretical explanations given above, the discussion continues by reporting the observed empirical results from a set of experiments in the next section.

4.2.1.1 Setting the parameters of RI: Empirical observations

Instead of a task-specific evaluation, the ability of RI-constructed vector spaces in preserving pairwise Euclidean distances is shown when the method's parameters are set differently.

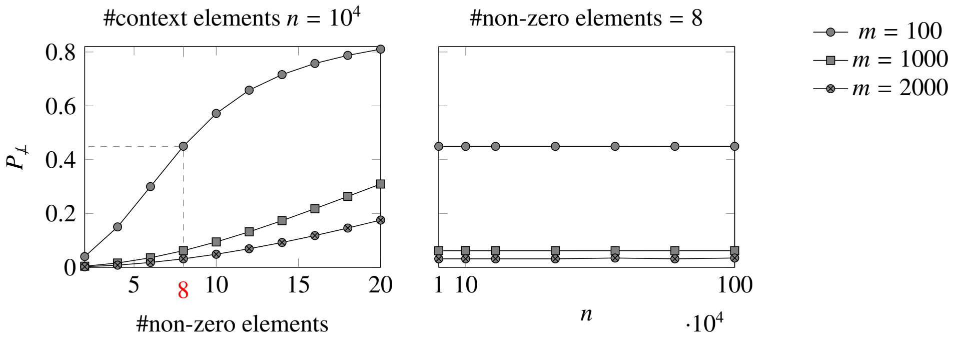


Figure 4.2: Orthogonality of index vectors: the y-axis shows the proportion of non-orthogonal pairs of index vectors (denoted by $P_{\not\perp}$) for sets of index vectors of various dimension $m = 100, 1000$, and 2000 obtained in a simulation. For sets of index vectors of a fixed size $n = 10^4$, the left figure shows the changes of $P_{\not\perp}$ when the number of non-zero elements increases. The right figure shows $P_{\not\perp}$ when the number of non-zero elements is fixed to 8, however, the number of index vector n increases. As shown in the figure, $P_{\not\perp}$ remains constant independently of n.

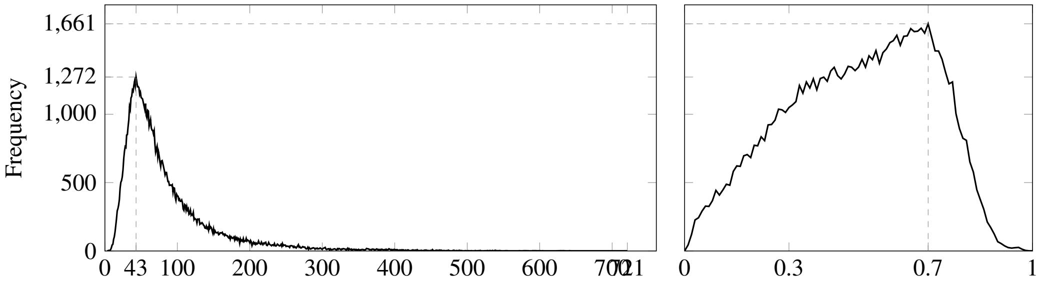
In the reported experiments, a subset of Wikipedia articles, which are chosen randomly from the *WaCkypedia_EN* corpus—that is, a 2009 dump of the English Wikipedia (Baroni et al., 2009)—are used.[1] A document-by-term VSM at its original high dimension is first constructed from a set of 10,000 articles (shown by D). A pre-processing of documents in D—that is, white-space tokenisation followed by the elimination of non-alphabetic tokens—results in a vocabulary of 192,117 terms. Each document in D is represented by a high-dimensional vector; each dimension represents an entry from the obtained vocabulary (as illustrated earlier in Figure 4.1). Therefore, the constructed VSM using this classic *one-dimension-per-context-element* method has a dimensionality of n = 192,117.[2]

To keep the experiments in a manageable size, each document d in D is randomly grouped by another 9 documents from D, which consequently gives 10,000 sets of a set of 10 documents. Using the constructed n-dimensional (n = 192,117) vector space, for each set of documents, the Euclidean distances between d and the remaining 9 documents in the set are computed. Subsequently, these 9 documents are sorted by their distance from d to obtain an ordered set of documents. The process therefore results in 10,000 ordered sets of 9 documents. The Euclidean distance is replaced with the cosine similarity and repeat the processes mentioned above. Figure 4.3 shows a histogram of the distribution of the distances between documents in these sets of documents. Figure 4.4 shows the distribution of the pairwise distances for all of the 10,000 documents; as shown, the distribution of the sampled distances closely resembles the distribution of all the pairwise distances in the model.

The procedure described above is repeated, however, by calculating distances in VSMs that are constructed using the RI method. Each term in the vocabulary is assigned to an

[1] The corpus can be obtained from http://wacky.sslmit.unibo.it/doku.php?id=corpora.

[2] In all the performed experiments, the frequency of terms in documents is used to indicate weights in corresponding vectors.



(a) Euclidean distance (b) Cosine

Figure 4.3: A histogram of the distribution of (a) the Euclidean distances and (b) the cosine similarities between pairs of vectors in the VSM of dimension 192,117 that are sampled randomly and employed for the experiments.

m-dimensional index vector and each document to a context vector. Context vectors are updated by accumulating index vectors to reflect the co-occurrences of documents and terms. Subsequently, the obtained context vectors are used to estimate the Euclidean distances and the cosine similarities between documents. The estimated distances are then used to create the ordered sets of documents, exactly as explained above. This process is repeated several times when the parameters of RI—that is, the dimension m and the number of non-zero elements in index vectors—are set to different values.

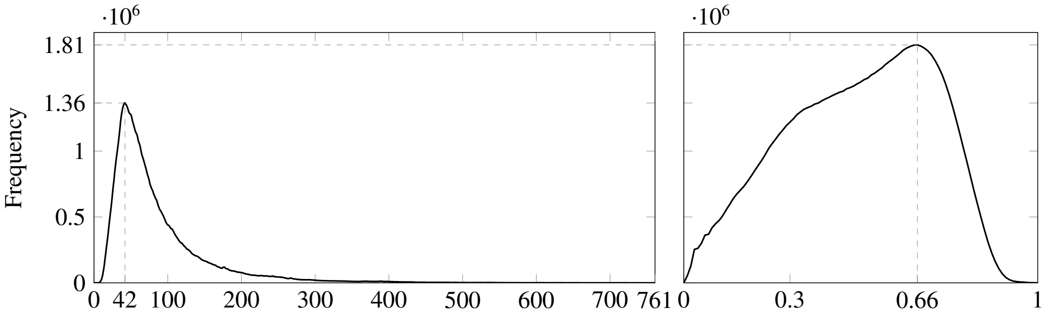
It is expected the relative Euclidean distances as well as the cosine similarities between documents in the RI-constructed VSMs to be the same as in the original high-dimensional VSM.[1] Hence, the ordered sets of documents obtained from estimated distances in the RI-constructed VSMs must be identical to the corresponding sets that are derived using the computed distances in the original high-dimensional VSM. For each VSM constructed using the RI method, therefore, the resulting ordered sets are compared with the obtained ordered sets from the original high-dimensional VSM using the Spearman's rank correlation coefficient measure (ρ).

The Spearman's rank correlation coefficient evaluates the strength of an association between two ranked variables, that is, two lists of sorted documents in our experiments. Given a list of sorted documents obtained from the original high-dimensional VSM (list_o) and its corresponding list obtained from a VSM constructed using the RI method (list_{RI}), Spearman's rank correlation for the two lists is given by

$$\rho = 1 - \frac{6\sum dif_i^2}{n(n^2-1)}, \tag{4.10}$$

where dif_i is the difference in paired ranks of documents in list_o and list_{RI}, and $n = 9$ is the number of documents that are sorted in each list. The average of ρ over the obtained sets

[1] The preservation of the cosine similarities can be verified mathematically, for example, see the provided proofs in Kaski (1998). Simply put, the cosine similarity can be expressed using the Euclidean distance when the length of vectors is normalised to unit length. This simple fact can be used to show that the cosine similarities are preserved when using the RI method.



(a) Euclidean distance

(b) Cosine

Figure 4.4: A histogram of the distribution of all the pairwise distances in the VSM of dimension 192,117 for (a) the Euclidean distances and (b) the cosine similarities.

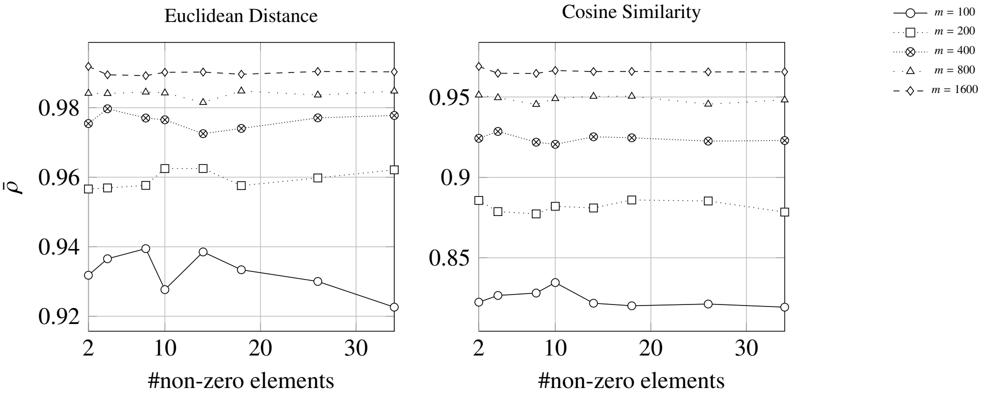
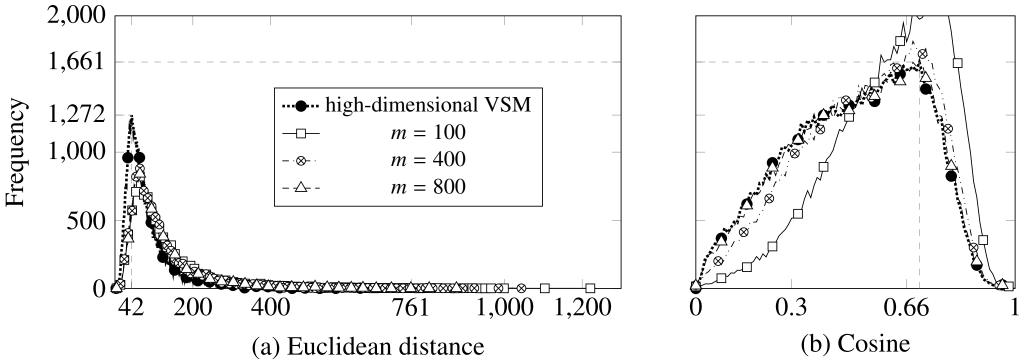


Figure 4.5: Correlation between the estimated Euclidean distances in RI-constructed vectors spaces and the original high-dimensional vector space: $\bar{\rho}$ shows the average of the Spearman's rank correlation coefficient between the ordered sets of documents that are obtained using the RI-constructed vectors spaces and the original high-dimensional vector space. Results are shown for both Euclidean distances as well as the cosine similarities when parameters of the RI method are set to different values.

of ordered set of documents ($\bar{\rho}$) is reported to quantify the performance of RI with respect to its ability to preserve ℓ_2-normed distances, when its parameters are set to different values: the closer $\bar{\rho}$ is to 1, the more similar the order of documents in an RI-constructed and the original high-dimensional VSM.

Figure 4.5 shows the obtained results. Since the dimension of the original vector space is very high, 2 non-zero elements per index vector are sufficient to construct a vector space that resembles relative distances between vectors in the original high-dimensional vector space, even for $m = 1600$. In addition, because only a small number of documents—that is, $p = 10000$—are modelled, even at the reduced dimension of $m = 100$, the estimated distances in the RI-constructed vector space shows a high correlation to the distances in the original vector space (i.e., $\bar{\rho} > 0.92$ for pairwise Euclidean distances and $\bar{\rho} > 0.82$ for the cosine similarity). As expected, the generated random baseline for $\bar{\rho}$ in Figure 4.5



(a) Euclidean distance (b) Cosine

Figure 4.6: Distribution of distances in the RI-constructed VSMs: as m increases, the distribution of the distances in the RI-constructed VSMs are becoming more similar to the distances' distribution in the original high-dimensional VSM.

is −0.002, that is, approximately 0. For $m = 1600$, the observed pairwise distances in the RI-constructed vector space are almost identical to the original vector space, that is, $\bar{\rho} > 0.99$ for Euclidean distances and $\bar{\rho} > 0.96$ for the cosine. Figure 4.6 compares the distribution of distances in the original high-dimensional VSM and the RI-constructed VSMs. As expected, when m increases, these distributions are becoming more similar to each other.

4.2.2 Related Work and Other Justifications of RI

As cited by Sahlgren (2005), the RI method was inspired from Kanerva's sparse distributed memory (SDM).[1] SDM, which was initially designed as a model of human long-term memory, is a cognitive-mathematical model. To formalise computation in several applications, it employs a high-dimensional *binary* vector space, the *Hamming* distance, as well as mathematical theorems that are often used in neural networks.[2] The RI method was then developed and justified by Kanerva et al. as an extension of SDM, without providing mathematical details, which are provided here. [3] An impression similar to Kanerva et al.'s (2000) RI can also be found in the methods suggested by Gallant (e.g., see Gallant,

[1] Perhaps more comprehensible than the JL lemma

[2] Recently, Snaider (2012, Chap. 2) has provided a summary of the SDM's mathematical foundation, and compared it with other mathematical models.

[3] Neither Sahlgren (2005) nor Kanerva et al. (2000) specify the proportion of the zero and non-zero elements in the index vectors, except that most of the elements of the index vectors are zero and only a *few* are 1 and −1. For instance, Kanerva et al. (2000) suggest 10 non-zero elements for a 4000-dimensional index vector without providing further explanation. Although Sahlgren and Karlgren (2005) suggest the following distribution (which can also be found in Sahlgren, 2006, chap. 4) for the elements of the index vectors:

$$r_{ij} = \begin{cases} +1 & \text{with probability } \frac{\beta/2}{m} \\ 0 & \text{with probability } \frac{m-\beta}{m} \\ -1 & \text{with probability } \frac{\beta/2}{m} \end{cases}, \tag{4.11}$$

1991).[1]

An account of random projection in Euclidean spaces similar to RI can be given following Kohonen's seminal work on self-organising maps (e.g., see Ritter and Kohonen, 1989, Appendix I). For instance, Kaski (1998) introduces *random mapping*, a dimension reduction technique that employs random projections in Euclidean spaces. Instead of the JL lemma, Kaski (1998) relies on the fact that the least distortion in a mapping in a Euclidean space, such as Equation 4.4, is attained when $\mathbf{R}$ is orthogonal. Using reported results in Hecht-Nielsen (1994), Kaski assumes that randomly created vectors are most likely to be orthogonal and suggests mapping by a random matrix constructed by i.i.d. random vectors $\mathbf{r} \sim \mathcal{N}_m(0, 1)$.[2] He then shows that the distortion in the *inner product* of pairs of vectors at reduced dimension is on average zero and its variance is less than $^2/_m$. Several other theorems and proofs, which give similar results to the JL lemma, can be found to explain the use of random projection for dimension reduction in Euclidean spaces in various applications (e.g., see Linial et al., 1995; Arriaga and Vempala, 2006).[3]

The viability of the random projection techniques in general, and the RI method specifically, have been verified in several research reports. Amongst them, experimental results reported by Bingham and Mannila (2001) admit that the dimension reduction using the suggested sparse random matrix in Achlioptas (2001) provides comparable results to the conventional dimension reduction techniques, such as truncated SVD, in a document similarity measurement application. In addition, a growing number of research in diverse application domains employ the RI technique for dimension reduction (e.e., see Jurgens and Stevens, 2009, 2010; Musto et al., 2012; Yannakoudakis and Briscoe, 2012).

Apart from setting the RI method's parameters, the proposed theorems in Section 4.2 enable us to (a) categorise methods employed for incremental VSM construction at a reduced dimensionality, and (b) provide mathematical justifications for several variations of the RI method proposed in research literature. First and foremost, incremental methods can be categorised based on the type of projections that they employ to construct VSMs at a reduced dimensionality (hence, the type of similarity metrics that they estimate). Despite that in natural language processing applications, the majority of these methods suggest the use of Gaussian random projections for estimating ℓ_2 norm-based similarities, a few researchers suggest random projections other than Gaussian to estimate similarities in VSMs other than ℓ_2-normed (e.g., see *TopSig* by Geva and De Vries (2011) and the random Manhattan indexing method proposed later in this chapter).

If a method based on random projections is employed to construct ℓ_2-normed VSMs,

they do not provide a criterion for choosing the values of m and β. The given distribution in Equation 4.11 expresses the probability of non-zero elements in terms of the dimension of the index vectors (i.e., m), and the number of non-zero elements (i.e., β). In this way, the degree of the sparsity of index vectors is shown by the probability of the non-zero elements.

[1] For an algorithmic description of these methods in a retrieval task see Caid and Oing (1997) and its references.

[2] Similar conclusion is drawn for the RI technique.

[3] Resulting from the popularity of *connectionist* methodology in late 80s and early 90s, the list of research that propose similar methodologies is very long. Giving a comprehensive view of this research effort is beyond the scope of this thesis. Interested readers can perhaps gain insight by following a citation network, for example, by starting from Pollack (1990) or any of the references listed in this section.

then its underlying mathematical principles is similar to RI[1]; hence, this method can be categorised in the same group of methods as RI. The major differences between methods in this category often result from (a) the procedure that they employ to construct a VSM at a reduced dimensionality (i.e., the second step of the RI procedure as explained from Equation 4.7 to 4.9) and/or (b) the weighting methodology that they employ in order to smooth collected co-occurrence frequencies.[2] The weighting process can be combined with the context vector construction, too.

As suggested earlier, the context vector construction can be carried out using a sequential scan of a corpus. The sequential scan, however, can be tailored to meet the requirements of a particular application. For example, context vectors can be updated every time the corpus is updated. Similarly, the weighting strategy can be changed to serve a specific purpose. Both of the alterations can take place by an intuitive or cognitive perspective, which may seem different from the RI technique. However, as long as substituted strategies can be interpreted using theorems suggested in Section 4.2, the resulting methods are, in essence, equivalent to the mapping that is given by the RI technique. In this case, the resulting vector space at reduced dimension still conforms to what is stated here for the RI-constructed VSMs.

The incremental semantic analysis (ISA) method, which is proposed by Baroni et al. (2007), and the reflective random indexing method, which is proposed by Cohen et al. (2010), are examples of the techniques discussed in the above paragraph. These methods offer interesting intuitions, other than the RI method, in order to enhance the results obtained for semantic similarity measurements in some applications. However, in both of these methods, the strategy employed for the construction of VSMs at reduced dimensionality can be interpreted as a technique for the adjustment of w_{ij} weights in Equation 4.4. Therefore, both methods are essentially the same as the RI method described here—that is, random projection with a sparse asymptotic Gaussian random matrix. For example, it can be verified that Baroni et al.'s (2007) ISA technique integrates a *Laplacian smoothing* to the RI's two-step procedure.

4.2.3 RI's Advantages Versus Limitations

The RI technique reduces the time and the space complexity of the required processes for constructing a VSM with regards to the values of

- n and m in Equation 4.4—that is, the original dimension of VSM and its reduced dimension obtained using RI, respectively;
- s in Equation 4.6—that is, the proportion of zero and non-zero elements in index vectors.

When using a sparse matrix representation, compared to a classic *one-dimension-per-context* VSM construction technique, the RI method imposes an additional $\beta - 1$ *addition* operations, where β is the number of non-zero elements in index vectors. However, this

[1] Or, can be equivalently represented as.

[2] A description of the weighting process in VSMs is given in Chapter 2).

additional computation is insignificant considering the fact that RI combines the construction of a vector space with the dimension reduction processes. RI eliminates the need for a resource-intensive dimension reduction technique, such as the truncated SVD. Evidently, by reducing the dimension of the vector space, RI enhances the time complexity of the process of measuring distances between vectors by an approximate factor of $\frac{n}{m}$. As suggested earlier, the use of sparse projections further enhances the time complexity of the construction of VSM by a factor equal to $\frac{1}{s}$, and, to an extent, the space complexity for storing and manipulating VSMs.

In many dimension reduction techniques other than random projection, the projection subspace is devised by the analysis of data in the original high-dimensional VSM. For instance, in order to employ truncated SVD, a linear equation that finds eigenvectors should be solved. Therefore, in these methods, if the structure of the data being analysed changes, the basis of the projection subspace also changes. Additionally, in such *data-sensitive* dimension reduction techniques, the vector space at the reduced dimension—thus, similarity assessments—is only available after the computation of the transformation and applying it to the data at the original high dimension. Both stipulations impose limitations when using a data-sensitive dimension reduction technique, which the RI method can resolve.

The first limitation is faced when updating a vector space that is followed by a data-sensitive method of dimensionality reduction. In this setup, updating the vector space results in cumbersome processes. The process of dimensionality reduction needs to be repeated in order to reflect the changes in the model. For example, the use of the truncated SVD demands the recalculation of the eigenvectors, and therefore the alternation of the transformation process, which affects all the vectors in the model at reduced dimension. As a result, a process such as distance computation should be repeated for all the vector space entries. However, in the RI technique, the employed subspace for dimension reduction, to a great extent, is independent of the data structure. Updating the vector space is carried out by the accumulation of existing or new index vectors, which affects only certain vectors. Thus, processes such as distance calculation are only necessary for the affected vectors.

The second limitation of a data-sensitive dimension reduction technique is that vector space at reduced dimension is available for processing only after the computation of the transformation. In contrast, when using the RI method, vector space at reduced dimension is available for processing during the construction of the vector space. As a result, similarity assessment is feasible at any time during the vector space construction, even when all the occurrences of entities in contexts are not observed. This is an extra advantage when processing frequently updated information, such as text streams in social media (e.g., see Sahlgren and Karlgren, 2009; Jurgens and Stevens, 2009; Karlgren et al., 2012).

The dimension of a vector space constructed using the RI method is fixed and, to a great extent, independent of the number of employed contexts and the size of corpus. However, the dimension of the vector space in a *one-dimension-per-context* model increases when new contexts are required to be added to the model. In a distributional model of semantics, due to the power-law distribution of context elements, appending a new entity to a model often requires appending new context elements to the model. The new entity most likely appears in/with context elements that have not yet appeared in the model. Therefore, in order to keep the model updated, its dimension should be increased to encompass new appended context elements. In contrast, in the RI technique, a large number of new context elements can be easily added to a vector space without changing its dimension, but at the expense of an insignificant loss of accuracy, which can be estimated by the JL lemma. A new context is defined and appended to the model simply by defining a new index vector.

The fixed dimensionality of the vector space constructed by RI and advance knowledge of its value are major advantages when dealing with big data, particularly in distributed computing frameworks. As described above, the induced vector space models using a technique such as the RI method scale up linearly with respect to the number of entities and not the number of contexts. In addition, the prior knowledge of the vectors' dimension is advantageous for load balancing in distributed computing frameworks (e.g., see Gufler et al., 2012, for an explanation of the load balancing problem).

The RI technique, however, comes with a number of limitations, which can be inferred from the proposed mathematical understanding of RI. The mathematical justification given in Section 4.2 explicitly states that the RI method, which employs a random matrix **R** whose elements are defined using the asymptotic distribution given in Equation 4.6, can only be applied for the approximation of similarity measures in the ℓ_2 normed spaces. That is, RI can be employed if similarity measures are derived from the ℓ_2 norm such as the Euclidean distance and the cosine similarity. For instance, the use of RI-constructed VSMs for estimating the city block distances between vectors—for example, as suggested in Lapesa and Evert (2013)—is not justified, at least mathematically.[1]

This list of the advantages and disadvantages is not exhaustive and new items can be added or removed according to the application context or the comparison framework.

4.2.4 A Summary of the Exposition's Outcomes

In Section 4.2, the use of Gaussian sparse random projections for dimension reduction in Euclidean spaces is described, which consequently arrives at the well-known random indexing technique. Accordingly, in Section 4.2.1.1, observed results in an empirical experiment are shown to understand the method's behaviour with respect to its ability to preserve pairwise Euclidean distances, or in general ℓ_2-normed-based similarity measures. In addition, several important outcomes from the mathematical description of the RI method are emphasised.

Firstly, whereas the original delineation of the method did not provide a concrete guideline for setting the method's parameters, Section 4.2.1 ameliorates the previous two-step procedure with criteria for choosing the dimensionality as well as proportion of zero and non-zero elements of index vectors.

Secondly, the proposed understanding of the RI method is employed to discern its limitations and application domain. It is proven that the employed random projections by the RI method do not preserve distances other than ℓ_2 (e.g., see Brinkman and Charikar,

[1] For example, see proofs in Brinkman and Charikar (2005). Also, see the reported empirical observations in Section 4.4.

2005). Hence, it is important to note that RI-constructed VSMs can only be used for estimating similarity measures that are derived from the ℓ_2 norm—for example, the Euclidean distance and the cosine similarity.

Thirdly, the rationale given in the aforementioned sections provides a framework to justify several variations of the RI technique mathematically. Although these methods are based on plausible intuitions, similar to RI, they lack theoretical justifications. For example, the given mathematical description can be employed to identify the method proposed in Baroni et al. (2007) as a variation of RI that employs *Laplacian* smoothing. Similarly, the same rationale can be used for categorisation of the methods that construct VSMs at a reduced dimensionality. This idea can be generalised to coordinate all other major processes that are often involved when using VSMs.

Lastly, the given understanding of the mechanism of RI can be employed to generalise RI to normed spaces other than ℓ_2. This generalisation can be achieved using random projections with a distribution other than asymptotic Gaussian—for example, as suggested in Indyk (2006); Li et al. (2013)—and altering Equation 4.6. Accordingly, in the next section, the random Manhattan indexing is proposed for constructing ℓ_1-normed VSMs incrementally and directly at a reduced dimensionality.

4.3 Random Projections in ℓ_1-Normed Space

As stated earlier, in a vector space, the similarity between vectors can be assessed using a *norm* structure. Besides the ℓ_2 norm, ℓ_1 norm is another *not so common* choice for the similarity measurement. The ℓ_1 norm for $\vec{v}$ is given by:

$$\|\vec{v}\|_1 = \sum_{i=1}^{n} |v_i|, \tag{4.12}$$

where |.| signifies the modulus.[1] Expectedly, a vector space endowed with the ℓ_1 norm is called an ℓ_1-normed space. The distance in an ℓ_1-normed vector space is often called the *Manhattan*, *taxicab*, or the *city block* distance. According to the definition given in Equation 4.1, the Manhattan distance between two vectors $\vec{v}$ and $\vec{u}$ is given by:

$$dist_1(\vec{v}, \vec{u}) = \|\vec{v} - \vec{u}\|_1 = \sum_{k=1}^{n} |v_i - u_j|. \tag{4.13}$$

Shown in Figure 4.7, the collection of the dash-dotted lines is the ℓ_1 distance between the two vectors. Similar to the ℓ_2-normed spaces, various normalisations of the ℓ_1 distance[2] define a family of ℓ_1-normed similarity metrics.

Similar to ℓ_2-normed spaces, the curse of dimensionality can obstruct efficient computation in ℓ_1 normed spaces. Both heuristic-based and transformation-based dimensionality reduction techniques can also be employed to alleviate the curse of dimensionality

[1] The definition of the norm is generalised to ℓ_p spaces with $\|\vec{v}\|_p = \left(\sum_i |v_i|^p\right)^{1/p}$; the discussion about ℓ_p-normed spaces other than $p = 1, 2$ goes beyond the scope of this thesis.

[2] As long as the axioms in the distance definition hold.

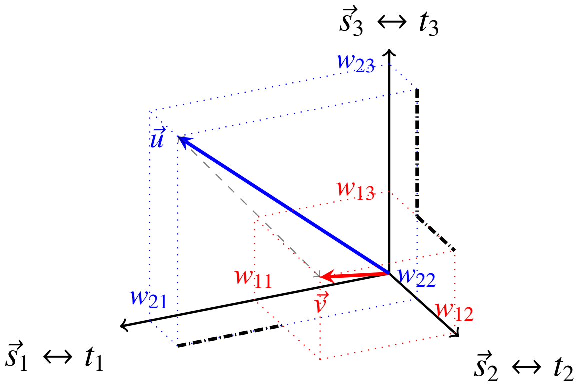

$\vec{s}_3 \leftrightarrow t_3$
w_{23}
$\vec{u}$
w_{13}
w_{11}
w_{22}
w_{21}
$\vec{v}$
w_{12}
$\vec{s}_1 \leftrightarrow t_1$
$\vec{s}_2 \leftrightarrow t_2$

Figure 4.7: The sum of the dash-dotted lines is the Manhattan distance between the two vectors $\vec{v_1} = (w_{11}, w_{12}, w_{13})$ and $\vec{v_2} = (w_{21}, w_{22}, w_{23})$. Whereas the Euclidean distance between the two vectors is the length of the straight line between them (the dashed line), the Manhattan distance between the two vectors is the sum of the absolute differences of their coordinates.

in ℓ_1-normed spaces. For example, similar to SVD truncation in ℓ_2-normed spaces, matrix factorisation techniques that guarantee the least distortion in the ℓ_1 distances can be employed (e.g., see Kwak, 2008). However, as discussed in Section 4.1, these methods are not desirable in a number of applications; for example, due to the resources they demand for computing VSMs at a reduced dimensionality, delays that they may cause in accessing VSMs at a reduced dimensionality, and frequent changes in the structure of data in VSMs. Accordingly, it is stated that random projections can be used to implement alternative dimensionality reduction techniques that can alleviate these problems.

In Euclidean spaces, random projections can be employed to introduce the RI technique. RI solves the problems stated above by combining the construction of a vector space and the dimensionality reduction process. Unlike methods that first construct a VSM at its original high dimension and conduct a dimensionality reduction afterwards, the RI method avoids the construction of the original high-dimensional VSM. Instead, it merges the vector space construction and the dimensionality reduction process. RI, thus, significantly enhances the computational complexity of deriving a VSM from text. However, the application of the RI technique (likewise, the standard truncated SVD in LSA) is limited to ℓ_2-normed spaces, that is, when similarities are assessed using a measure based on the ℓ_2 distance. It is verified that using RI causes large distortions in the ℓ_1 distances between vectors (Brinkman and Charikar, 2005). Hence, the RI technique is not suitable for constructing VSMs if similarities are computed using the ℓ_1 distance.

Depending on the distribution of vectors in a VSM, the performance of similarity measures based on the ℓ_1 and the ℓ_2 norms varies from one task to another. For instance, it is suggested that the ℓ_1 distance is more robust to the presence of outliers and non-Gaussian noise than the ℓ_2 distance (see the problem description in Ke and Kanade, 2003)). Hence, the use of the ℓ_1 distance can be more reliable than the ℓ_2 distance in certain applications. For instance, Weeds et al. (2005) suggest that the ℓ_1 distance outperforms other similarity metrics in a term classification task. In another experiment, Lee (1999) observed that the ℓ_1 distance gives more desirable results than the cosine and the

ℓ_2 measures.

In this section, a novel method called *random Manhattan indexing* (RMI) is introduced, which employs random projections in ℓ_1-normed spaces. RMI constructs a VSM directly at a reduced dimension while it preserves the pairwise ℓ_1 distances between vectors in the original high-dimensional VSM. A computationally enhanced version of RMI called *random Manhattan integer indexing* (RMII) is then introduced. RMI and RMII, using the similar principles employed by RI, merge the construction of a VSM and dimension reduction into an incremental—thus, efficient and scalable—process. In Section 4.3.1, the RMI method is explained and evaluated. In Section 4.3.2, the RMII method is explained. RMI and RMII are compared to RI in Section 4.4.

4.3.1 Random Manhattan Indexing

In this section, the Random Manhattan Indexing (RMI) method is proposed: an algorithm that adapts random projections in order to introduce an incremental procedure for constructing ℓ_1-normed vector spaces at a reduced dimensionality. The RMI method employs a two-step procedure: (a) the creation of *index vectors* and (b) the construction of *context vectors*.

In the first step, each context element is assigned exactly to one *index vector* $\vec{r_i}$. Index vectors are high-dimensional and generated randomly such that entries r_j of index vectors have the following distribution:

$$r_i = \begin{cases} \frac{-1}{U_1} & \text{with probability } \frac{s}{2} \\ 0 & \text{with probability } 1 - s \\ \frac{1}{U_2} & \text{with probability } \frac{s}{2} \end{cases}, \tag{4.14}$$

where U_1 and U_2 are independent uniform random variables in $(0, 1)$. In the second step, each target linguistic entity that is being analysed in the model is assigned to a context vector $\vec{v_c}$ in which all the elements are initially set to 0. For each encountered co-occurrence of a linguistic entity and a context element—for example, through a sequential scan of an input corpus—$\vec{v_c}$ that represents the linguistic entity is accumulated by the index vector $\vec{r_i}$ that represents the context element—that is, $\vec{v_c} = \vec{v_c} + \vec{r_i}$. This process results in a VSM of a reduced dimensionality that can be used to estimate the ℓ_1 distances between linguistic entities.

In the constructed VSM by RMI, the ℓ_1 distance between vectors is given by the *sample median* Indyk (2000). For given vectors $\vec{v}$ and $\vec{u}$, the approximate ℓ_1 distance between vectors is estimated by

$$\hat{L}_1(\vec{u}, \vec{v}) = \text{median}\{|v_i - u_i|, i = 1, \cdots, m\}, \tag{4.15}$$

where m is the dimension of the VSM constructed by RMI, and $|.|$ denotes the modulus.

Similar to RI, RMI employs random projections (RPs): a high-dimensional VSM is mapped onto a random subspace of lowered dimension expecting that—with a high probability—relative distances between vectors are approximately preserved. As suggested earlier in Equation 4.4, using the matrix notation, this projection is given by

$$\mathbf{M}'_{p\times m} = \mathbf{M}_{p\times n} \times \mathbf{R}_{n\times m}, \quad m \ll p, n, \tag{4.16}$$

where **R** is often called the *random projection matrix*, and **M** and **M′** denote p vectors in the original n-dimensional and reduced m-dimensional vector spaces, respectively.

In RMI, the stated mapping in Equation 4.16 is given by *Cauchy random projections*. Indyk (2000) suggests that vectors in a high-dimensional space $\mathbb{R}^n$ can be mapped onto a vector space of lowered dimension $\mathbb{R}^m$ while the relative pairwise ℓ_1 distances between vectors are preserved with a high probability. In Indyk (2000, Theorem 3) and Indyk (2006, Theorem 5), it is shown that for an $m \geq m_0 = \log(1/\delta)^{O(1/\epsilon)}$, where $\delta > 0$ and $\epsilon \leq 1/2$, there exists a mapping from $\mathbb{R}^n$ onto $\mathbb{R}^m$ that guarantees the ℓ_1 distances between any pair of vectors $\vec{u}$ and $\vec{v}$ in $\mathbb{R}^n$ after the mapping does not increase by a factor more than $1 + \epsilon$ with constant probability δ, and it does not decrease by more than $1 - \epsilon$ with probability $1 - \delta$.

In Indyk (2000), this projection is proved to be obtained using a random projection matrix **R** that has a *Cauchy distribution*—that is, for r_{ij} in **R**, $r_{ij} \sim C(0, 1)$. Since **R** has a Cauchy distribution, for every two vectors $\vec{u}$ and $\vec{v}$ in the high-dimensional space $\mathbb{R}^n$, the projected differences $x = \hat{\vec{u}} - \hat{\vec{v}}$ also have Cauchy distribution, with the scale parameter being the ℓ_1 distances:

$$x \sim C(0, \sum_{i=1}^{n} |u_i - v_i|). \tag{4.17}$$

As a result, in Cauchy random projections, estimating the ℓ_1 distance between any two vectors $\vec{u}$ and $\vec{v}$ boils down to the estimation of the Cauchy scale parameter from i.i.d. samples x. Because the expectation value of x is infinite,[1] the sample mean cannot be employed to estimate the Cauchy scale parameter. Simply put, this means that $\sum_{i=1}^{n} |u_i - v_i|$ can be used to estimate distances at the reduced dimensionality. Instead, using the 1-stability of Cauchy distribution, Indyk (2000) proves that the median can be employed to estimate the Cauchy scale parameter, and thus the ℓ_1 distances at the projected space $\mathbb{R}^m$.

Subsequent studies simplified the method proposed by Indyk (2000). Particularly, Li (2007) shows that **R** with Cauchy distribution can be substituted by a *sparse* **R** that has a mixture of symmetric 1-Pareto distribution. A 1-Pareto distribution can be sampled by $1/U$, where U is an independent uniform random variable in $(0, 1)$. This results in a random matrix **R** that has the same distribution as described by Equation 4.14.

The RMI's two-step procedure is explained using the basic properties of matrix arithmetic and the descriptions given above. Given the projection in Equation 4.16, the first step of RMI refers to the construction of **R**: index vectors are the row vectors of **R**. The second step of the process refers to the construction of **M′**: context vectors are the row vectors of **M′**. Using the distributive property of multiplication over addition in matrices,[2] it can be verified that the explicit construction of **M** and its multiplication to **R** can be substituted by a number of summation operations, exactly as explained from Equation 4.7 to Equation 4.9 for projections in Euclidean spaces. That is, **M** can be represented by the sum of unit vectors in which a unit vector corresponds to the co-occurrence of a linguistic entity and a context element. The result of the multiplication of each unit vector and **R** is

[1] That is, $E(x) = \infty$, since x has a Cauchy distribution. Cauchy distribution is a heavy tailed distribution, therefore, the expected value does not exist.

[2] That is, $(\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC}$.

the row vector that represents the context element in **R**—that is, the index vector. Therefore, $\mathbf{M}'$ can be computed by the accumulation of the row vectors of **R** that represent encountered context elements, as stated in the second step of the RMI procedure.

4.3.1.1 Alternative distance estimators

As stated above, Indyk (2000) suggests using the sample median for the estimation of the ℓ_1 distances. However, Li (2008) argues that sample median estimator can be biased and inaccurate, particularly if the targeted reduced dimensionality (i.e., m) is small. Hence, Li (2008) suggests using the geometric mean estimator instead of the median sample.[1] Accordingly, the ℓ_1 distances at the reduced dimensionality can be estimated by

$$\hat{L}_1(\vec{u}, \vec{v}) = (\prod_{i=1}^{m} |u_i - v_i|)^{\frac{1}{m}}. \tag{4.18}$$

I suggest computing the $\hat{L}_1(\vec{u}, \vec{v})$ in Equation 4.18 using the arithmetic mean of logarithm-transformed values of $|u_i - v_i|$. Therefore, with the help of the logarithmic identities, the multiplications and the exponent power in Equation 4.18 are, respectively, transformed to a sum and a multiplication:

$$\hat{L}_1(\vec{u}, \vec{v}) = \exp\Big(\frac{1}{m} \sum_{i=1}^{m} \ln(|u_i - v_i|)\Big). \tag{4.19}$$

For a computational implementation, Equation 4.19 for estimating $\hat{L}_1$ is more plausible than Equation 4.18—for example, the overflow is less likely to happen during the process. Moreover, calculating the median involves sorting an array of real numbers. Thus, computation of the geometric mean in logarithmic scales can be faster than computation of the median sample, particularly when the value of m is large.

4.3.1.2 RMI's parameters

In order to employ the RMI method for the construction of an ℓ_1-normed VSM at a reduced dimensionality, two model parameters should be decided: (a) the targeted reduced dimensionality of the VSM, which is indicated by m in Equation 4.16 and (b) the number of non-zero elements in index vectors, which is determined by s in Equation 4.14. In contrast to the classic *one-dimension-per-context-element* methods of VSM construction and similar to RI,[2] the value of m in RPs and thus in RMI is chosen independently of the number of context elements in the model (n in Equation 4.16).

In RMI, m determines the probability and the maximum expected amount of distortions ϵ in the pairwise distance between vectors. Based on the proposed refinements of Indyk (2000, Theorem 3) by Li et al. (2007), it is verified that the pairwise ℓ_1 distance between any p vectors is approximated within a factor $1 \pm \epsilon$, if $m = O(\log p/\epsilon^2)$, with a constant probability. Therefore, the value of ϵ in RMI is subject to the number of vectors

[1] See also Li et al. (2007, Lemma 5–9).

[2] That is, n context elements are modelled in an n-dimensional VSM.

p in the model. For a fixed p, a larger m yields to lower bounds on the distortion with a higher probability. Because a small m is desirable from the computational complexity outlook, the choice of m is often a trade-off between accuracy and efficiency. Similar to discussions in Section 4.2.1 for RI, m can be seen as the capacity of the model for accommodating new vectors without causing a large amount of distortion in the distances between vectors.[1] According to my experimental experiences, $m \geq 400$ is suitable for most applications.

The number of non-zero elements in index vectors, however, is decided by the number of context elements (i.e., n) and the sparseness of the VSM at its original dimension (denoted by β). Li (2007) suggests $\frac{1}{\mathrm{O}(\sqrt{\beta n})}$ as the value of s in Equation 4.14. As discussed elsewhere, because of the long tail distribution of context elements and linguistic entities (e.g., the Zipfian distribution of words in documents), VSMs employed in distributional semantics—and in general, text analysis—are highly sparse. The sparsity of a VSM in its original dimension (i.e., β) is often considered to be around $10^{-4} \leq \beta \leq 10^{-2}$. However, as the original dimension of VSM n is very large—otherwise there would be no need for dimensionality reduction—the index vectors are often very sparse. Similar to m, larger s produces smaller errors. However, during the construction of a VSM, a large s imposes more processes than a small s.

It is important to note that the influence of s in RI and RMI is different. Whereas in RI, a large s may cause further distortion in the relative estimated distances, in RMI a larger s can help the estimated relative distances converge faster to the relative distances in the original high-dimensional space. Based on the performed experiments and without providing mathematical proofs, for an m-dimensional VSM, I suggest $2\lceil\frac{m}{2\sqrt{\alpha n}}\rceil$ non-zero elements, in which half of them are positive and the other half are negative.

4.3.1.3 Empirical evaluation of RMI

This section reports the performance of the RMI method with respect to its ability to preserve the relative ℓ_1 distance between linguistic entities in a VSM—similar to the observations reported earlier to evaluate RI.[2] Therefore, instead of a task-specific evaluation, it is shown that the relative ℓ_1 distance between a set of words in a high-dimensional *word-by-document* model remains intact when the model is constructed at a reduced dimensionality using the RMI technique. This evaluation is repeated for a *document-by-word* model using the same dataset used in Section 4.2.1.1 for RI, too. The effect of various settings of the RMI's parameters are then explored in the observed results.

The purpose of the reported evaluations is to show the ability of RMI in preserving the relative ℓ_1 distances. Depending on the structure of the data that is being analysed and the objective of the task in hand, the performance of the ℓ_1 distance for similarity measurement can be better or worse than other similarity metrics (e.g., see the experiments in Bullinaria and Levy, 2007). The evaluation designed in this section takes this fact into the consideration. Hence, the purpose of the reported evaluations is not to show the superiority of RMI (thus the ℓ_1 distance) to dimensionality reduction techniques in normed

[1] Li et al. (2007) details the choice of m using mathematical arguments and observations over synthesised date.

[2] See the experiment in Section 4.2.1.1.

PoS	**Words**						
Noun	website	email	support	software	students	skills	project
	research	nhs	link	services	organisations		
Adjective	online	digital	mobile	sustainable	global	unique	excellent
	disabled	new	current	fantastic	innovative		
Verb	use	visit	improve	provided	help	ensure	develop

Table 4.1: Words employed in the experiments. These words are the chosen examples in Ferraresi et al. (2008).

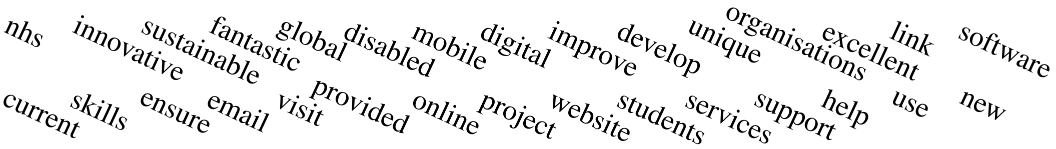


Figure 4.8: List of words sorted by their ℓ_1 distance to the word *research*. The distance increases from left to right and top to bottom.

spaces other than ℓ_1 (e.g., RI or truncated SVD in ℓ_2-normed spaces) in a specific task. If, in a task, the ℓ_1 distance shows higher performance than the ℓ_2 distance, then the RMI technique is preferable to the RI technique or truncated SVD. Contrariwise, if the ℓ_2 norm shows higher performance than the ℓ_1 norm, then RI or truncated SVD are more desirable than the RMI method.

In the reported experiment, a word-by-document model is first constructed from ukWaC at its original high dimension. UkWaC is a freely available corpus of 2,692,692 web documents, nearly 2 billion tokens and 4 million types (Baroni et al., 2009).[1] Therefore, a word-by-document model constructed from this corpus using the classic one-dimension-per-context-element method has the maximum dimension of 2.69 million. In order to keep the experiments computationally tractable, the reported results are limited to 31 words from this model, which are listed in Table 4.1. Figure 4.9 shows the increase in the dimensionality of the VSM when a new word from this list is added to the VSM.

In the designed experiment, a word from the list is taken as the reference and its ℓ_1 distance to the remaining 30 words is calculated using the vector representations in the high-dimensional VSM. These 30 words are then sorted in ascending order by the calculated ℓ_1 distance. The procedure is repeated for all of the 31 words in the list, one by one. Therefore, the procedure results in 31 sorted lists, each containing 30 words. Figure 4.8 shows an example of such an obtained sorted list, in which the reference is the word *research*.[2]

The procedure described above is replicated to obtain the lists of sorted words from

[1] UkWaC can be obtained from http://wacky.sslmit.unibo.it/doku.php?id=corpora.

[2] Please note that the number of possible arrangements of 30 words without repetition in a list in which the order is important (i.e., all permutations of 30 words) is 30!. As a result, the probability of generating the same sorted list of words when they are arranged by their ℓ_1 distance to another word is $\frac{1}{30!}$.

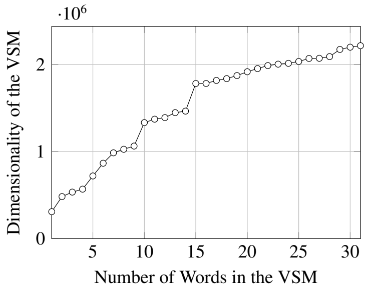


Figure 4.9: The increase in the dimensionality of a word-by-document model constructed from the ukWaC: Adding a new word to the model causes the VSM's dimension to burst when it is constructed using the classic one-document-per-dimension.

VSMs that are constructed at reduced dimensionality using the RMI technique, when the method's parameters—that is, the dimension of index vectors as well as the proportion of zero and non-zero elements in them—are set differently. It is expected the obtained relative ℓ_1 distances between each reference word and the 30 other words in an RMI-constructed VSM to be the same as the obtained relative distances in the original high-dimensional VSM. Therefore, for each VSM that is constructed by the RMI technique, the resulting sorted lists of words are compared by the sorted lists that are obtained from the original high-dimensional VSM.

Similar to the other experiments reported in this chapter, the Spearman's rank correlation coefficient (ρ) is employed to compare the sorted lists of words and thus the degree of distance preservation in the RMI-constructed VSMs at reduced dimensionality. Hence, given a list of sorted words obtained from the original high-dimensional VSM (list$_o$) and its corresponding list obtained from a VSM of reduced dimensionality (list$_{RMI}$), the Spearman's rank correlation for the two lists is calculated using Equation 4.10 (in which, dif_i is the difference in paired ranks of words in list$_o$ and list$_{RMI}$, and $n = 30$ is the number of words in each list). The average of ρ over the 31 lists of sorted words, denoted by $\bar{\rho}$, is reported to indicate the performance of RMI with respect to its ability for distance preservation. The closer $\bar{\rho}$ is to 1, the better the performance of RMI with respect to the relative ℓ_1 distance preservation.

Figure 4.10 shows the observed results at a glance when the distances are estimated using the median (Equation 4.15). As shown in the figure, when the dimension of the VSM is above 400 and the number of non-zero elements is more than 12, the obtained relative distances from the VSMs constructed by the RMI technique start to be analogous to the relative distances that are obtained from the original high-dimensional VSM, that is, a high correlation ($\bar{\rho} > 0.90$). For the baseline, the average correlation of $\bar{\rho}_{random} = -0.004$ between the sorted lists of words obtained from the high-dimensional VSM and 31×1000 lists of sorted words that are obtained by randomly assigned distances is reported.

Figure 4.11 shows the same results as Figure 4.10, however, in minute detail and only for VSMs of dimension $m \in \{100, 400, 800, 3200\}$. In these plots, squares (■) indicate the $\bar{\rho}$ while the error bars show the best and the worst observed ρ amongst all the sorted lists of words. The minimum value of the ρ-axis is set to 0.611, which is the worst observed

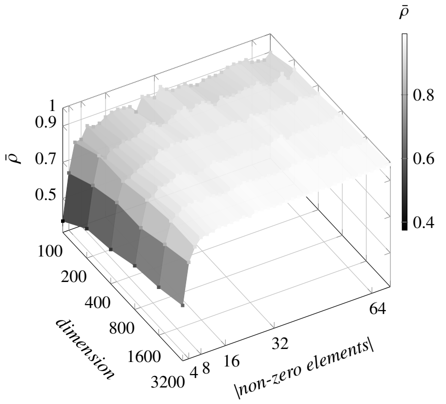


Figure 4.10: The $\bar{\rho}$ axis shows the observed average Spearman' rank correlation between the order of the words in the lists that are sorted by the ℓ_1 distance obtained from the original high-dimensional VSM and the VSMs that are constructed by RMI at reduced dimensionality using index vectors of various numbers of non-zero elements.

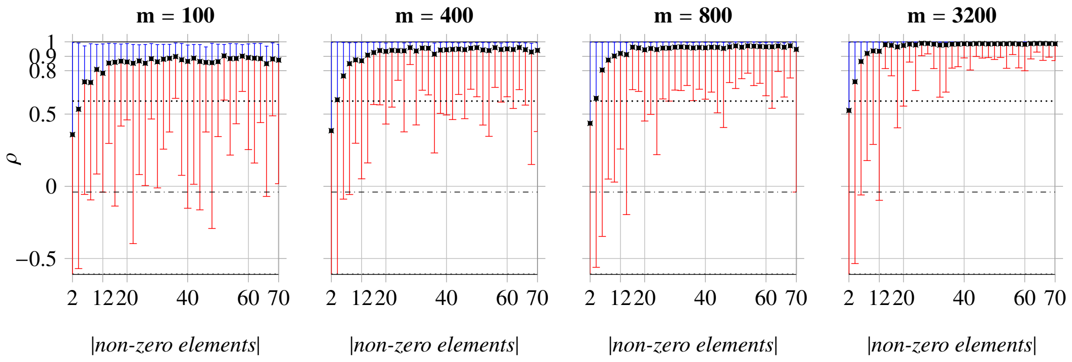


Figure 4.11: Detailed observation of the obtained correlation between relative distances in RMI-constructed VSMs and the original high-dimensional VSM. The ℓ_1 distance is estimated using the median. The squares denote $\bar{\rho}$ and the error bars show the best and the worst observed correlations. The dashed-dotted line shows the random baseline.

correlation in the baseline (i.e., randomly generated distances). The dotted line (i.e., $\rho = .591$) shows the best observed correlation in the baseline and the dashed-dotted line shows the average correlation in the baseline ($\rho = -0.004$). As suggested in Section 4.3.1.2, it can be verified that an increase in the dimension of VSMs (i.e., m) increases the stability of the obtained results (i.e., the probability of preserving distances increases). Therefore, for large values of m (i.e., $m > 400$), the difference between the best and the worst observed ρ decreases; average correlation $\bar{\rho} \to 1$, and the relative distances in RMI-constructed VSMs become identical to those in the original high-dimensional VSM.

Figure 4.12 represents the obtained results in the same setting as above, however, when the distances are approximated using the geometric mean (Equation 4.19). The obtained average correlations $\bar{\rho}$ from the geometric mean estimations are almost identical to

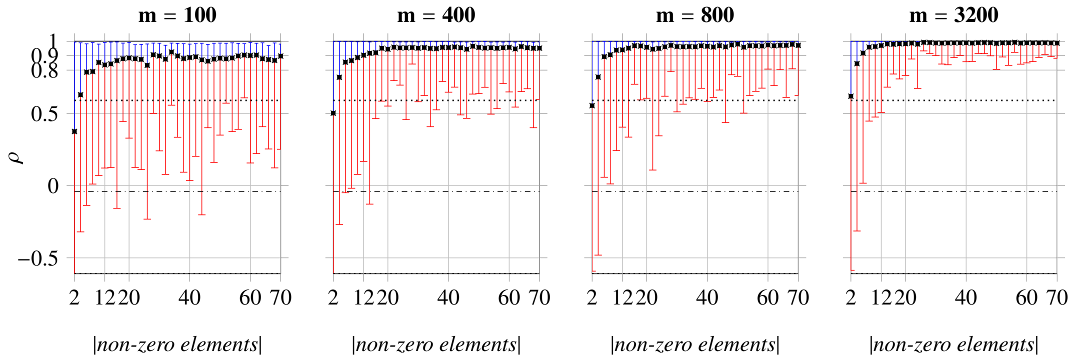


Figure 4.12: The observed results when the ℓ_1 distance in RMI-constructed VSMs is estimated using the geometric mean.

the median estimations. However, as expected, the geometric mean estimations are more reliable for small values of m; particularly, when using the geometric mean, the worst observed correlations are higher than those observed when using the median estimator.

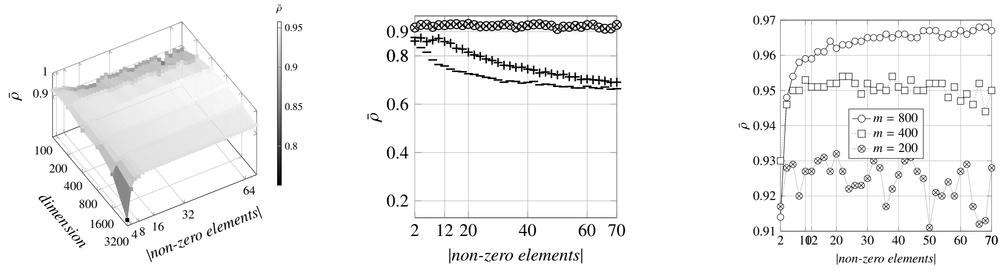
This experiment is also repeated over the document-by-word models that have been employed earlier in Section 4.2.1.1. Instead of the Euclidean distance, however, the constructed models are used to verify the ability of RMI-constructed VSMs to preserve ℓ_1 distances between vectors. Results are shown in Figure 4.13.

4.3.2 Random Manhattan Integer Indexing

The application of the RMI method is hindered by two obstacles: float arithmetic operations required for the construction and processing of the RMI-constructed VSMs and the calculation of the product of large numbers when ℓ_1 distances are estimated using the geometric mean.

The proposed method for the generation of index vectors in RMI results in index vectors of non-zero elements that are real numbers. Consequently, index vectors and thus context vectors are arrays of floating point numbers. These vectors must be stored and accessed efficiently when the RMI technique is employed in an application. However, storing and processing floating numbers are resource intensive, and therefore not desirable in real-world applications—particularly when dealing with large corpora. Even if the requirement for the storage of index vectors is alleviated—for example, using a derandomisation technique for their generation—context vectors that are derived from these index vectors are still arrays of float numbers and their storage and process is of high space and time complexity.

To tackle this problem, I suggest substituting the value of non-zero elements of RMI's index vectors (given in Equation 4.14) from $\frac{1}{U}$ to integer values of $\lfloor \frac{1}{U} \rfloor$, where $\lfloor \frac{1}{U} \rfloor \neq$



(a) (b) (c)

Figure 4.13: The RMI's ability to preserve relative ℓ_1 distances in a document-by-word model: The performance is assessed using the observed $\bar{\rho}$ over a set of 10,000 documents chosen randomly from the *WaCkypedia_EN* in an experiment similar to Section 4.2.1.1. Figure 4.13a shows the overall observed result when the RMI' parameters are set differently. Figure 4.13b shows the same results only when the dimension of VSM is 200. In this figure, the minimum value of the $\bar{\rho}$-axis is set to the best observed correlation $\rho = 0.1375$ when distances are generated randomly (first baseline). The + and − marks show $\bar{\rho}$ when ℓ_1 distance is estimated in RI-constructed VSMs of dimensionality 1600 using the estimator in Equation 4.19 and the standard definition of the ℓ_1 distance, respectively. Figure 4.13c plots the same observed results only for RMI and when $m \in \{200, 400, 800\}$. These results are similar to the experiments with the word-by-document model. It can be verified that an increase in the dimension of VSM results in an increase in $\bar{\rho}$.

0—that is:

$$r_i = \begin{cases} \lfloor \frac{1}{U_1} \rfloor & \text{with probability } \frac{s}{2} \\ 0 & \text{with probability } 1 - s \\ \lfloor \frac{1}{U_2} \rfloor & \text{with probability } \frac{s}{2} \end{cases} . \tag{4.20}$$

I argue that the resulting random projection matrix still has an asymptomatic Cauchy distribution. Therefore, the proposed methodology to estimate the ℓ_1 distance between vectors is still valid. The ℓ_1 distance between context vectors must be still estimated using either the median or the geometric mean.

The use of the median estimator—for the reasons stated in Section 4.3.1.1—is not plausible. On the other hand, the computation of the geometric mean can be laborious as the overflow is highly likely to happen during its computation. Using the value of $\lfloor \frac{1}{U} \rfloor$ for non-zero elements of index vectors, it is evident that for any pair of context vectors $\vec{u} = (u_1, \cdots, u_m)$ and $\vec{v} = (v_1, \cdots, v_m)$, if $u_i \neq v_i$ then $|u_i - v_i| \geq 1$. Therefore, for $u_i \neq v_i$, $\ln |u_i - v_i| \geq 0$ and thus $\sum_{i=1}^{m} \ln(|u_i - v_i|) \geq 0$. In this case, the exponent in Equation 4.19 is a scale factor that can be discarded without a change in the relative distances between vectors.[1] Based on the intuition that the distance between a vector and itself is zero and the explanation given above, inspired by smoothing techniques and without being able to provide mathematical proofs, I suggest estimating the relative distances between vectors

[1] Please note that according to the axioms in the distance definition, the distance between two numbers is always a non-negative value. When index vectors consist of non-zero elements of real numbers, the value of $|u_i - v_i|$ can be between 0 and 1, that is, $0 < |u_i - v_i| < 1$. Therefore, $\ln(|u_i - v_i|)$ can be a negative number and thus the exponent scale is required to make sure that the result is a non-negative number.

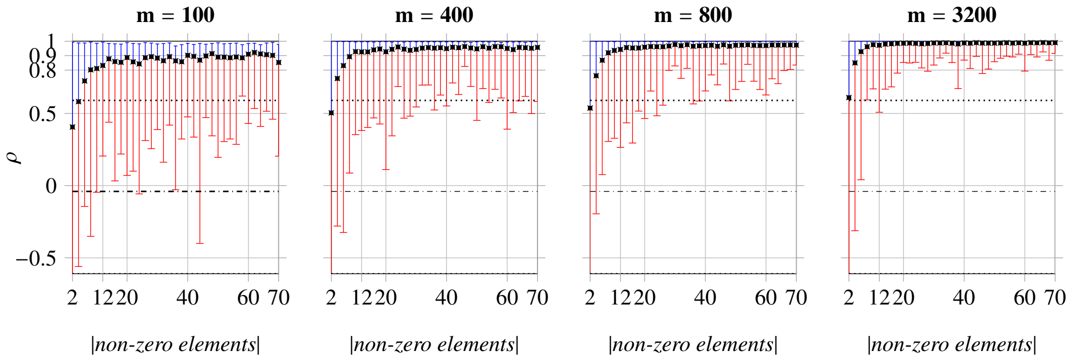


Figure 4.14: The observed results when using the RMII method for the construction and estimation of the ℓ_1 distances between vectors. The method is evaluated in the same setup as the RMI technique.

using

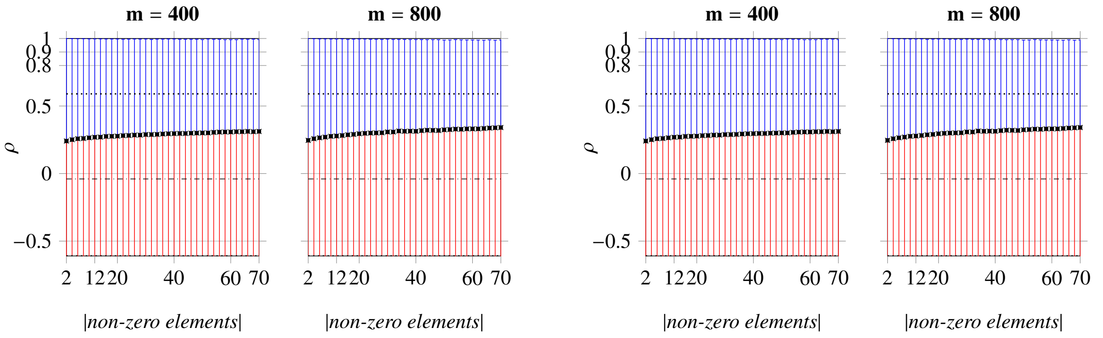
$$\hat{L}_1(\vec{u}, \vec{v}) = \sum_{\substack{i=1 \\ u_i \neq v_i}}^{m} \ln(|u_i - v_i|). \tag{4.21}$$

In order to distinguish the above changes in RMI, the resulting technique is called random Manhattan integer indexing (RMII). The experiment described in Section 4.3.1.2 is repeated using the RMII method. As shown in Figure 4.14, the obtained results are almost identical to the observed results when using the RMI technique. While RMI performs slightly better than RMII in lower dimensions—for example, $m = 400$—RMII shows more stable behaviour than RMI at higher dimensions—for example $m = 800$. However, in all these cases, RMII demands less memory and processing resources for its computations.

4.4 Comparing RMI and RI

RMI and RI utilise a similar two-step procedure consisting of the creation of index vectors and the construction of context vectors. In addition, both RMI and RI are incremental techniques that construct a VSM at reduced dimensionality directly, without requiring the VSM to be constructed at its original high dimension. Despite these similarities, RMI and RI are motivated by different applications and mathematical theorems. RMI is justified using asymptotic Cauchy random projections whereas RI is justified using asymptotic Gaussian random projections.

As described above, RMI approximates the ℓ_1 distance using a *non-linear estimator*, which has not yet been employed for the construction of VSMs and the calculation of ℓ_1 distances in distributional approaches to semantics. In contrast, RI approximates the ℓ_2 distance using a *linear estimator*. RI has initially been justified using the mathematical model of the sparse distributed memory (SDM). Later, as suggested in this chapter, the RI method was explained using the lemma proposed by Johnson and Lindenstrauss



(a) The standard ℓ_1 distance definition

(b) The median estimator

Figure 4.15: Evaluation of RI for estimating ℓ_1 distances for $m = 400$ and $m = 800$ when the distances are calculated using (a) the standard definition of distance in ℓ_1-normed spaces and (b) the median estimator. The obtained results using RI do not show a correlation to the ℓ_1 distances in the original high-dimensional VSM.

(1984)—which elucidates random projections in Euclidean spaces (see Section 4.2 for details). Although both the RMI and RI methods can be established as α-stable random projections—respectively for $\alpha = 1$ and $\alpha = 2$—the methods cannot be compared as they address different goals. If, for a given task, the ℓ_1 norm outperforms the ℓ_2 norm, then RMI is preferable to RI. Contrariwise, if the ℓ_2 norm outperforms the ℓ_1 norm, then RI is preferable to RMI. As implied in the reported evaluations and stated above, RI and RMI cannot be replaced with each other. As shown in the previous sections, using RI for dimensionality reduction causes a large distortion in the relative ℓ_1 distances between vectors. Reversely, RMI does not preserve the relative ℓ_2 distances between vectors.

To support the earlier claim that RI-constructed VSMs cannot be used for the ℓ_1 distance estimation, the RI method is evaluated in the experimental setup that has been used for the evaluation of RMI and RMII. In these experiments, however, RI is employed to construct vector spaces at reduced dimensionality and estimate the ℓ_1 distance using Equation 4.13 (the standard ℓ_1 distance definition) and Equation 4.15 (the median estimator) for $m \in 400, 800$. As shown in Figure 4.15, the experiments support this claim.

4.5 Summary

In this chapter, the applications of random projections for constructing vector spaces with reduced dimensionality are outlined. As discussed, these methods can be employed to enhance the performance in distributional semantic models.

This chapter has two contributions in particular. First, in Section 4.2, the random indexing method is explained mathematically; and its two-step procedure is delineated using sparse asymptotic Gaussian random projections. Consequently, criteria for setting the method's parameters are suggested. Second, in Section 4.3, a novel technique, named random Manhattan indexing (RMI), for the construction of ℓ_1-normed VSMs directly at reduced dimensionality is suggested. In addition, Section 4.3.2 introduces the random Manhattan integer indexing (RMII) technique—that is, a computationally enhanced version of the RMI technique. The ability of these methods to preserve ℓ_1 distances are demonstrated using empirical evaluations.

As discussed, the use of random projections in the incremental way suggested in this chapter has a number of benefits. First, it enhances the computational complexity of the construction of models by combining the process of collecting co-occurrences with the dimensionality reduction process. The result is a vector space model constructed directly with reduced dimensionality. Second, because of the reduced dimensionality of the vectors, the subsequent similarity computations are performed faster. Third, the proposed incremental method provides the capability of updating a model at any time during its use, which makes it suitable for frequently updated data, particularly, in the context of big-text data analytics.

As suggested in Section 4.4, vector spaces that are constructed using random projections, such as the RI and RMI techniques, are limited to the specific normed space that they are designed for. There are methods that claim they can overcome this restriction—for example, Li et al.'s (2006a) conditional random sampling. However, they have not yet been applied to the vector space models of semantics. The use of these methods is one way to extend the presented research in this chapter. In the proposed methods in this chapter, only one random projection is applied before estimating distances between vectors. However, it is possible to use a chain of projections—for example, as it is used in the implementations of neural network algorithms. Such combinations are also possible for RMI and RI.

Last but not least, the design principles employed in this chapter to reintroduce RI and propose RMI and RMII can be employed for normed spaces other than the ℓ_1 and the ℓ_2-normed. This is an exciting future research that has not yet been investigated for natural language processing applications. Random projections are a vibrant research topic in modern mathematics and statistics and the future advances in these fields will most definitely result in new efficient methods and techniques for big text data analytics.

Reference List

Achlioptas, D. (2001). Database-friendly random projections. In *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 274–281, Santa Barbara, CA, USA. ACM. 110, 117

Arriaga, R. and Vempala, S. (2006). An algorithmic theory of learning: Robust concepts and random projection. *Machine Learning*, 63(2):161–182. 117

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky Wide Web: A collection of very large linguistically processed Web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226. 113, 127

Baroni, M., Lenci, A., and Onnis, L. (2007). ISA meets Lara: An incremental word space model for cognitively plausible simulations of semantic learning. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 49–56, Prague, Czech Republic. Association for Computational Linguistics. 118, 121

Bingham, E. and Mannila, H. (2001). Random projection in dimensionality reduction: Applications to image and text data. In *KDD '01: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 245–250, New York, NY, USA. ACM. 117

Brand, M. (2006). Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra and its Applications*, 415(1):20–30. Special Issue on Large Scale Linear and Nonlinear Eigenvalue Problems. 108

Brinkman, B. and Charikar, M. (2005). On the impossibility of dimension reduction in L1. *Journal of the ACM*, 52(5):766–788. 120, 122

Bullinaria, J. A. and Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526. 126

Caid, W. R. and Oing, P. (1997). System and method of context vector generation and retrieval. 117

Cohen, T., Schvaneveldt, R., and Widdows, D. (2010). Reflective random indexing and indirect inference: A scalable method for discovery of implicit connections. *Journal of Biomedical Informatics*, 43(2):240–256. 118

Dasgupta, S. and Gupta, A. (2003). An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures and Algorithms*, 22(1):60–65. 110

Ferraresi, A., Zanchetta, E., Baroni, M., and Bernardini, S. (2008). Introducing and evaluating ukWaC, a very large web-derived corpus of English. In Evert, S., Kilgarriff, A., and Sharoff, S., editors, *Proceedings of the 4th Web as Corpus Workshop (WAC-4): Can we beat Google?*, pages 47–54. 127

Gallant, S. I. (1991). A practical approach for representing context and for performing word sense disambiguation using neural networks. *Neural Computation*, 3(3):293–309. 116

Geva, S. and De Vries, C. M. (2011). TOPSIG: Topology preserving document signatures. In Berendt, B., de Vries, A., Fan, W., Macdonald, C., Ounis, I., and Ruthven, I., editors, *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, pages 333–338, Glasgow, Scotland, UK. ACM. 117

Gufler, B., Augsten, N., Reiser, A., and Kemper, A. (2012). Load balancing in MapReduce based on scalable cardinality estimates. In *ICDEW'12: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering Workshops*, pages 522–533, Virginia, USA. IEEE Computer Society. 120

Hecht-Nielsen, R. (1994). Context vectors: General purpose approximate meaning representations self-organized from raw data. In *Computational Intelligence: Imitating Life*, pages 43–56. IEEE Press. Papers presented at the 1994 World Congress on Computational Intelligence (WCCI) held in summer in Orlando, Florida. 117

Indyk, P. (2000). Stable distributions, pseudorandom generators, embeddings and data stream computation. In *Proceedings: 41st Annual Symposium on Foundations of Computer Science*, pages 189–197, Redondo Beach, California. IEEE Computer Society. 123, 124, 125

Indyk, P. (2006). Stable distributions, pseudorandom generators, embeddings, and data stream computation. *Journal of the ACM*, 53(3):307–323. 121, 124

Johnson, W. and Lindenstrauss, J. (1984). Extensions of Lipschitz mappings into a Hilbert space. In Beals, R., Beck, A., Bellow, A., and Hajian, A., editors, *Conference on Modern Analysis and Probability (1982: Yale University)*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society. 109, 110, 132

Jurgens, D. and Stevens, K. (2009). Event detection in blogs using temporal random indexing. In Constantin Orasan, L. H. and Forascu, C., editors, *Proceedings of the Workshop on Events in Emerging Text Types*, pages 9–16, Borovets, Bulgaria. Association for Computational Linguistics. 117, 119

Jurgens, D. and Stevens, K. (2010). HERMIT: Flexible clustering for the SemEval-2 WSI task. In *SemEval 2010: 5th International Workshop on Semantic Evaluation:*

Proceedings of the Workshop, pages 359–362, Uppsala, Sweden. Association for Computational Linguistics. 117

Kanerva, P. (1993). Sparse Distributed Memory and related models. In Hassoun, M. H., editor, *Associative neural memories: theory and implementation*, chapter 3, pages 50–76. Oxford University Press, New York, NY, USA. 109

Kanerva, P., Kristoferson, J., and Holst, A. (2000). Random indexing of text samples for latent semantic analysis. In Gleitman, L. R. and Josh, A. K., editors, *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, page 1036, Mahwah, New Jersey. Erlbaum. 108, 109, 112, 116

Karlgren, J., Sahlgren, M., Olsson, F., Espinoza, F., and Hamfors, O. (2012). Profiling reputation of corporate entities in semantic space: Notebook for RepLab at CLEF 2012. In *CLEF (Online Working Notes/Labs/Workshop)*. 119

Kaski, S. (1998). Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *The 1998 IEEE International Joint Conference on Neural Networks Proceedings: IEEE Worl Congress on Computational Intelligence*, volume 1, pages 413–418, Alaska, USA. 114, 117

Ke, Q. and Kanade, T. (2003). Robust subspace computation using ℓ_1 norm. Technical Report CMU-CS-03-172, Carnegie Mellon University. 122

Kwak, N. (2008). Principal component analysis based on L1-norm maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1672–1680. 122

Lapesa, G. and Evert, S. (2013). Evaluating neighbor rank and distance measures as predictors of semantic priming. In *Proceedings of the Fourth Annual Workshop on Cognitive Modeling and Computational Linguistics*, pages 66–74, Sofia, Bulgaria. Association for Computational Linguistics. 120

Lee, L. (1999). Measures of distributional similarity. In *37th Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, pages 25–32, Maryland, USA. Association for Computational Linguistics. 122

Li, P. (2007). Very sparse stable random projections for dimension reduction in l_α $(0 \leq \alpha \leq 2)$ norm. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 440–449, San Jose, US. ACM. 124, 126

Li, P. (2008). Estimators and tail bounds for dimension reduction in ℓ_α $(0 \leq \alpha \leq 2)$ using stable random projections. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 10–19, CA, USA. Association for Computing Machinary and Society for Industrial and Applied Mathematics. 125

Li, P., Church, K. W., and Hastie, T. J. (2006a). Conditional random sampling: A sketch-based sampling technique for sparse data. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 873–880. MIT Press, Cambridge, MA. 134

Li, P., Hastie, T. J., and Church, K. W. (2006b). Very sparse random projections. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 287–296, New York, NY, USA. ACM. 110, 112

Li, P., Hastie, T. J., and Church, K. W. (2007). Nonlinear estimators and tail bounds for dimension reduction in L_1 using Cauchy random projections. *Journal of Machine Learning Research*, 8:2497–2532. 125, 126

Li, P., Samorodnitsk, G., and Hopcroft, J. (2013). Sign Cauchy projections and chi-square kernel. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 26*, pages 2571–2579. Curran Associates, Inc. 121

Linial, N., London, E., and Rabinovich, Y. (1995). The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245. 117

Lupu, M. (2014). On the usability of random indexing in patent retrieval. In Hernandez, N., Jäschke, R., and Croitoru, M., editors, *Graph-Based Representation and Reasoning*, volume 8577 of *Lecture Notes in Computer Science*, pages 202–216. Springer International Publishing. 111

Matoušek, J. (2008). On variants of the Johnson-Lindenstrauss lemma. *Random Structures and Algorithms*, 33(2):142–156. 110

Musto, C., Narducci, F., Lops, P., Semeraro, G., Gemmis, M., Barbieri, M., Korst, J., Pronk, V., and Clout, R. (2012). Enhanced semantic TV-show representation for personalized electronic program guides. In Masthoff, J., Mobasher, B., Desmarais, M., and Nkambou, R., editors, *User Modeling, Adaptation, and Personalization*, volume 7379 of *Lecture Notes in Computer Science*, pages 188–199. Springer Berlin Heidelberg. 117

Polajnar, T. and Clark, S. (2014). Improving distributional semantic vectors through context selection and normalisation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 230–238, Gothenburg, Sweden. Association for Computational Linguistics. 111

Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence*, 46(1–2):77–105. 117

QasemiZadeh, B. (2015a). *Investigating the Use of Distributional Semantic Models for Co-Hyponym Identification in Special Corpora*. PhD thesis, National University of Ireland, Galway. i

QasemiZadeh, B. (2015b). Random indexing revisited. In Biemann, C., Handschuh, S., Freitas, A., Meziane, F., and Metais, E., editors, *Natural Language Processing* *and Information Systems*, volume 9103 of *Lecture Notes in Computer Science*, pages 437–442. Springer International Publishing. 105

QasemiZadeh, B. and Handschuh, S. (2015). Random indexing explained with high probability. In Kral, P. and Matousek, V., editors, *Text, Speech and Dialogue (TSD)*, volume 9302 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 480–489, Pilsen, Czech. Springer International Publishing Switzerland. 105

Ritter, H. and Kohonen, T. (1989). Self-organizing semantic maps. *Biological Cybernetics*, 61(4):241–254. 117

Sahlgren, M. (2005). An introduction to random indexing. Technical report, Swedish ICT (SICS). Retrived from https://www.sics.se/~mange/papers/RI_intro.pdf. 108, 109, 112, 116

Sahlgren, M. (2006). *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD thesis, Stockholm University. 116

Sahlgren, M. and Karlgren, J. (2005). Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Natural Language Engineering*, 11(3):327–341. 116

Sahlgren, M. and Karlgren, J. (2009). Terminology mining in social media. In *CIKM'09: Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 405–414, Hong Kong, China. ACM. 119

Salton, G., Wong, A., and Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620. 107

Snaider, J. (2012). *Integer Sparse Distributed Memory and Modular Composite Representation*. PhD thesis, Computer Science. 116

Stein, B. (2007). Principles of hash-based text retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 527–534, Amsterdam, The Netherlands. ACM. 112

Weeds, J., Dowdall, J., Schneider, G., Keller, B., and Weir, D. (2005). Using distributional similarity to organise BioMedical terminology. *Terminology*, 11(1):3–4. 122

Yannakoudakis, H. and Briscoe, T. (2012). Modeling coherence in ESOL learner texts. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 33–43, Montreal, Canada. Association for Computational Linguistics. 117

Zadeh, B. Q. and Handschuh, S. (2014a). Random Manhattan indexing. In *25th International Workshop on Database and Expert Systems Applications (DEXA)*, pages 203–208, Munich, Germany. IEEE. 105

Zadeh, B. Q. and Handschuh, S. (2014b). Random Manhattan integer indexing: Incremental L1 normed vector space construction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1713–1723, Doha, Qatar. Association for Computational Linguistics. 105