Random Manhattan Indexing

Behrang Q. Zadeh
Insight Centre for Data Analytics
National University of Ireland, Galway
Email: behrang.qasemizadeh@insight-centre.org

Siegfried Handschuh
Insight Centre for Data Analytics
National University of Ireland, Galway
Email: siegfried.handschuh@insight-centre.org

***Abstract*—Vector space models (VSMs) are mathematically well-defined frameworks that have been widely used in text processing. In these models, high-dimensional, often sparse vectors represent text units. In an application, the similarity of vectors—and hence the text units that they represent—is computed by a distance formula. The high dimensionality of vectors, however, is a barrier to the performance of methods that employ VSMs. Consequently, a dimensionality reduction technique is employed to alleviate this problem. This paper introduces a new method, called Random Manhattan Indexing (RMI), for the construction of ℓ_1 normed VSMs at reduced dimensionality. RMI combines the construction of a VSM and dimension reduction into an incremental, and thus scalable, procedure. In order to attain its goal, RMI employs the sparse Cauchy random projections.**

I. Introduction

Prior to its processing, natural language text must be converted into a format that is suitable for the method that processes it. Vector space is an algebraic structure that is often employed to serve this purpose. Each text unit being analyzed—such as words, phrases or documents—is represented as a vector in a high-dimensional vector space. Each dimension of this vector space expresses a particular characteristic of the text units. These characteristics constitute[1] statistical information about the usage of the text units in certain contexts, depending on the objective of the task in hand. The result is a mathematically well-defined model, known as a vector space model (VSM). VSMs are often employed by methods that deal with the meaning of text units, with renowned application in distributional approaches to semantics [1].

In a VSM, a distance formula defines the similarity between vectors. Hence, the relative proximity of vectors to one another interprets the meaning of text units that they represent. As the number of text units that are being modelled in a VSM increases, the number of contexts that are required to be utilized to capture their meaning escalates. This phenomenon is explained using power-law distributions of text units in contexts. For example, Zipf's law states that most words are rare, while few words are used frequently. As a result, extremely high-dimensional vectors, which are also sparse—i.e. most of the elements of the vectors are zero—represent text units. The high dimensionality of the vectors results in setbacks, which are colloquially known as *the curse of dimensionality*. Therefore, a dimensionality reduction technique is often employed to alleviate these problems.

In this paper, we introduce a novel technique called *Random Manhattan Indexing* (RMI). RMI merges the construction of a VSM and dimension reduction into an incremental, and thus efficient and scalable, process. The proposed method is similar to the Random Indexing (RI) technique [2,3] and *TopSig* [4,5]. RMI, however, is the counterpart of these methods for ℓ_1 normed vector spaces. This paper describes the method and its underlying theory. Section II recalls the basics of VSMs. Section III briefly reviews dimensionality reduction techniques. The RMI method is explained in Section IV. In Section V, we report the performance of the RMI method in an experimental evaluation. We conclude in Section VI.

II. Preliminaries

In a VSM, a collection of p text units whose meanings are analyzed using n context elements builds a subspace of an n-dimensional vector space V_n consisting of p vectors. In this model, the vector $\vec{s}_i$ in the standard basis of V_n (for $1 \leq i \leq n$)[2] represents the i^{th} context element. A text unit is denoted by a vector $\vec{v}$ that can be expressed by a linear combination of $\vec{s}_i$

$$\vec{v} = w_1\vec{s}_1 + \cdots + w_n\vec{s}_n, \tag{1}$$

where w_i is a real number that determines the association of the text unit to the i^{th} context. The coordinate of $\vec{v}$, i.e. $(w_1, \cdots, w_n)$, thus shows the correlations between the text unit and the context elements in the model. A matrix $\mathbf{M}_{p\times n}$ of real numbers—in which rows and columns denote text units and context elements, respectively—can be employed to represent the coordinates of vectors in V_n.

Salton's document-by-term model in information retrieval (IR) is the most familiar example of a VSM [6]. Given a number of documents, i.e. text units, and n distinct terms, i.e. context elements, each document d is represented by an n-dimensional vector $\vec{d} = (w_1, \cdots, w_n)$, where w_i is a numeric value that associates the document d to the term t_i. In this model, each dimension, i.e. $\vec{s}_i$ in Equation 1, represents a term (Fig. 1). In this document-by-term model, the values of w may correspond to the frequency of each of the terms in each of the documents to implement the *bag of words* hypothesis: it is assumed the relevance of documents can be assessed by counting terms that they share, regardless of their order or syntactic usage patterns. Therefore, documents with similar vectors are expected to have similar meaning. In IR applications, queries are also treated as pseudo-documents; hence, the comparison of vectors provides a method to resolve retrieval tasks. Likewise, VSMs can implement hypotheses other than the bag of words in order to address processing text units other than documents, in contexts other than term occurrences. In all these VSMs, however, the distance between vectors measures the similarities between text units.

[1] Usually, but not necessarily.

[2] That is, informally, the dimensions of the vector space.

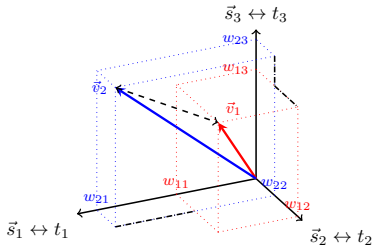


Fig. 1. Illustration of a *document-by-term* model consisting of 2 documents and 3 terms. Each element of the standard basis s_i, i.e. each dimension, represents each one of the 3 terms in the model. The 3-dimensional vectors $\vec{v_1} = (w_{11}, w_{12}, w_{13})$ and $\vec{v_2} = (w_{21}, w_{22}, w_{23})$ represent the two documents in the model. The dashed line shows the Euclidean distance between the two vectors, while the sum of dash-dotted lines is the Manhattan distance between them.

In order to assess the similarity between vectors, V is endowed with a structure called a *norm*. A norm $\|.\|$ is a function that satisfies certain axioms and maps vectors from V to the set of non-negative real numbers, i.e. $V \mapsto [0, \infty)$. The pair of $(V, \|.\|)$ is then called a *normed vector space*. In a normed space, the distance between vectors is defined by a function that satisfies certain axioms and assigns a real value to each pair of vectors:

$$d : V \times V \mapsto \mathbb{R}, \quad d(\vec{v}, \vec{t}) = \|\vec{v} - \vec{u}\|. \tag{2}$$

Subsequently, the similarity between vectors can be assessed by their distances: the smaller the distance between two vectors, the more similar they are.

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—also called the Euclidean norm—of a vector $\vec{v} = (v_1, \cdots, v_n)$ is defined as $\|\vec{v}\|_2 = \sqrt{\sum_{i=1}^{n} v_i^2}$. Using the given definitions for the distance in Equation 2 and the ℓ_2 norm, the Euclidean distance is measured as $d_2(\vec{v}, \vec{u}) = \|\vec{v} - \vec{u}\|_2 = \sqrt{\sum_{i=1}^{n} (v_i - u_i)^2}$. In the ℓ_2 normed vector spaces, various similarity metrics are defined using different normalization of the Euclidean distance between vectors. For example, the cosine similarity between vectors is a similarity measure in the ℓ_2 normed spaces, which is defined by the Euclidean distance between vectors when their length/norm is normalized to unity.

The similarity between vectors can also be computed in ℓ_1 normed spaces.[3] The ℓ_1 norm for $\vec{v}$ is given by $\|\vec{v}\|_1 = \sum_{i=1}^{n} |v_i|$. The distance in ℓ_1 normed vector spaces is often called the *Manhattan* or the *city block* distance. According to the definition given in Equation 2, the Manhattan distance between two vectors $\vec{v}$ and $\vec{u}$ is given by $d_1(\vec{v}, \vec{u}) = \|\vec{v} - \vec{u}\|_1 = \sum_{i=1}^{n} |v_i - u_i|$. Similar to the ℓ_2 spaces, various normalizations of the ℓ_1 distance[4] define a family of ℓ_1 normed similarity metrics. Depending on the distribution of vectors, the performance of the ℓ_1 and ℓ_2 similarity measures varies from one task to another. The ℓ_1 distance is more robust to the presence of outliers and non-Gaussian noise than the ℓ_2 distance (e.g. see [7]). Hence, the ℓ_1 distance can be more reliable than the ℓ_2 distance in certain applications, e.g. see [8].

[3]The definition of the norm is generalized to ℓ_p spaces with $\|\vec{v}\|_p = \left(\sum_{i=1}^{\infty} |v_i|^p\right)^{1/p}$, which is beyond the scope of this paper.

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

III. Dimension Reduction

The curse of dimensionality is a common barrier in the application of VSMs. When the number of context elements increases, the dimension of VSMs increases and, thus, hinders the computation of distances between vectors. In a large number of text-based analysis applications, e.g. distributional models of semantics, the number of context elements often varies as a power of the number of text units: a phenomenon known as *power-law* (or *heavy-tailed*) distribution. For instance, a document collection contains a huge number of terms that are only shared between a few of the documents. Many of these terms are irrelevant to the content of a document. Therefore, in a document-by-term model, adding a new document collection to the model entails appending a large set of terms. Consequently, the large number of terms results in high dimensionality of the VSM; few common terms between documents results in sparseness of the vectors and the presence of irrelevant terms introduces noise.

Dimension reduction, which usually follows the construction of a VSM, alleviates the problems listed above by reducing the number of context elements that are employed for the construction of the VSM. In its simple form, dimension reduction is performed as a *selection process*: choose a subset of context elements and eliminate the rest using a heuristic. Alternatively, *transformation* methods can be employed. A transformation method maps a vector space V_n onto a V_m of lowered dimension, i.e. $\tau : V_n \mapsto V_m$, $m \ll n$. The vector space at reduced dimension, i.e. V_m, is often the best approximation of the original V_n in a *sense*.

One category of transformation methods employ matrix factorization techniques. In this group of methods, truncated singular value decomposition (SVD), which is often identified by the latent semantic indexing technique in IR [9], is a well-known example. Matrix factorization methods such as truncated SVD are *data-sensitive*: if the structure of the data being analyzed changes, i.e. when either the text units or context elements are updated, e.g. some are removed or new ones are added, the transformation needs to be recomputed and reapplied to the whole VSM to reflect the updates. For example, when using truncated SVD in a document-by-term model, a change in either the document collection or the terms demands the recalculation of the SVD. In addition, in these methods, 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. Therefore, the VSM at reduced dimension is available for processing only after the whole sequence of these processes. Construction of the VSM at its original dimension is computationally expensive and a delay in access to the VSM at reduced dimension is not desirable. Therefore, these methods are not suitable in several applications, particularly when dealing with frequently updated big text-data such as applications in the web context.

A family of dimensionality reduction techniques addresses the above-mentioned problems of the matrix factorization-based methods using the principles of *random projections* (RP). In RP, a high-dimensional vector space is mapped onto a random subspace of lowered dimension expecting that—with a high probability—relative distances between vectors are approximately preserved. Hence, RP avoids the high compu-

tational complexity of the matrix factorization process. Using the matrix notation, this projection can be given by

$$\mathbf{M}'_{p\times m} = \mathbf{M}_{p\times n} \times \mathbf{R}_{n\times m}, \quad m \ll p, n, \tag{3}$$

where $\mathbf{R}$ is often called the *random matrix*, and $\mathbf{M}$ and $\mathbf{M}'$ denote p vectors in the original n-dimensional and reduced m-dimensional vectors spaces, respectively. Unlike methods that first construct a VSM at its original high dimension and conduct a dimensionality reduction afterwards, a category of RP-based methods—such as TopSig and RI as well as its variants, e.g. [10]—avoid the construction of the original high-dimensional VSM. Instead, using the *distributive property of matrix multiplication*, these methods combine the construction of a vector space and the dimensionality reduction process (i.e. the right-hand side of Equation 3) to generate a VSM directly at reduced dimension (i.e. $\mathbf{M}'$ in Equation 3). As a result, these methods significantly enhance the computational complexity of deriving a VSM of lowered dimensionality from text.

The procedure of the construction of a VSM at reduced dimension (i.e. $\mathbf{M}'_{p\times m}$) in these methods is best described by the two-step procedure in the RI algorithm: (a) the creation of *index vectors* and (b) the construction of *context vectors* [3]. In the first step, each context element is assigned exactly to one index vector. An index vector is high dimensional and randomly generated; most of the elements are 0 and only a few are set to 1 and -1. In the second step, each target text unit is assigned to a vector, called a context vector. Context vectors have the same dimension as index vectors have, and all of their elements, initially, are set to 0. For each encountered co-occurrence of a text unit and a context element—e.g. through a sequential scan of an input text collection—the context vector $\vec{v_c}$ that represents the text unit is accumulated by the index vector $\vec{r_i}$ that represents the context element, i.e. $\vec{v_c} = \vec{v_c} + \vec{r_i}$. The result is $\mathbf{M}'$ that represents the text units at reduced dimension. As can be inferred, the first step refers to the construction of $\mathbf{R}$: index vectors are the row vectors of $\mathbf{R}$. And, the second step refers to the computation of $\mathbf{M} \times \mathbf{R}$.

For example, in the construction of a document-by-term model using the RI method, each 'term' is assigned exactly to one index vector. In the second step, each 'document' is allocated to a context vector. The context vector of a document is then updated by the accumulation of the index vectors of all the terms that occurred in the document. The first and the second step of the process can be arranged in sequences different to that described here. For example, new terms can be added to the model at any time by defining new index vectors. To reflect the change in the model, the context vectors of documents that contain newly added terms should be updated by the accumulation of the newly added index vectors. Similarly, for adding a new document to the model, a new context vector is created and will be added to the VSM.

RI employs a random projection matrix $\mathbf{R}$ that has independent and identically distributed (i.i.d) entries r_i such that

$$r_i = \begin{cases} -1 & \text{with probability } \frac{s}{2} \\ 0 & \text{with probability } 1-s \\ 1 & \text{with probability } \frac{s}{2} \end{cases}, \tag{4}$$

where s determines the number of non-zero elements. Using this information and the mathematical proofs given in [11] and [12], it can be verified that RI is a RP technique for Euclidean spaces—i.e. ℓ_2 normed. In Euclidean spaces, RPs are elucidated using the Johnson–Lindenstrauss lemma (JL lemma) [13].

In the original proof of the JL lemma, $\mathbf{R}$ is an orthogonal matrix, and the lemma, thus, is proved for an orthogonal projection. However, the computation of an orthogonal matrix is difficult. Subsequent studies simplified the method by showing that an orthogonal $\mathbf{R}$ can be replaced by a randomly generated matrix of standard Gaussian distribution (see [14] for proofs and references). Particularly, [11] and [12] show that $\mathbf{R}$ can be a matrix with the asymptotic distribution described in Equation 4. Therefore, using $\mathbf{R}$ with the stated asymptotic distribution in Equation 4 is only valid for RPs in the ℓ_2 normed spaces. It has been proved that using these projections causes large distortions in the ℓ_1 distance between vectors [15]. Hence, if the similarities are computed using the ℓ_1 distance, then RI and other techniques that are based on the JL lemma are not suitable for the VSM construction. The proposed RMI technique extends the presented idea to the ℓ_1 normed spaces.

IV. Random Manhattan Indexing

We propose the Random Manhattan Indexing (RMI) method: a novel technique for the construction of ℓ_1 normed vector spaces at reduced dimensionality. RMI is motivated by *Cauchy random projections*. Theorem 3 in [16] suggests an embedding for ℓ_1 normed spaces similar to the one that is proposed by the JL lemma for ℓ_2 normed spaces. It is shown that for an $m \geq m_0 = \log(1/\delta)^{\mathrm{O}(1/\epsilon)}$, where $\delta > 0$ and $\epsilon \leq 1/2$, there exists a mapping from a real vector space $\mathbb{R}^n$ onto $\mathbb{R}^m$, $m \ll n$, that guarantees the ℓ_1 distance between any pair of vectors 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 $1-\delta$. This projection is proved to be obtained using a random matrix $\mathbf{R}$ that has a *Cauchy distribution*—i.e. for $r_{ij} \in \mathbf{R}$, $r_{ij} \sim C(0,1)$. Since $\mathbf{R}$ has a Cauchy distribution, for every two vectors $\vec{u}$ and $\vec{v}$ in $\mathbb{R}^n$, the projected differences $x = \hat{\vec{u}} - \hat{\vec{v}}$ in $\mathbb{R}^m$ also have Cauchy distribution, with the scale parameter being the ℓ_1 distances, i.e. $x \sim C(0, \sum_{i=1}^{n} |u_i - v_i|)$. As a result, in Cauchy random projections, estimating the ℓ_1 distances boils down to the estimation of the Cauchy scale parameter from i.i.d. samples x. Because the expected value of the Cauchy random variable does not exist (i.e. infinite), [16] suggests using the *sample median* in order to estimate the ℓ_1 distance between vectors. Subsequent research improved the proposed projection in [16]. In [17], it is shown that $\mathbf{R}$ with *Cauchy distribution* can be substituted by a sparse $\mathbf{R}$ that has a mixture of symmetric 1-Pareto distribution. In [18], it is shown that the ℓ_1 distance can be estimated using non-linear estimators such as the geometric mean other than the sample median.

Accordingly, RMI is a two-step procedure. First, each context element is assigned exactly to one index vector. Index vectors are generated randomly such that entries r_i of index vectors have the following distribution:

$$r_i = \begin{cases} \frac{-1}{U_1} & \text{with probability } \frac{s}{2} \\ 0 & \text{with probability } 1-s \\ \frac{1}{U_2} & \text{with probability } \frac{s}{2} \end{cases}, \tag{5}$$

where U_1 and U_2 are independent uniform random variables in (0,1). The second step of RMI is identical to that in RI and TopSig. Each text unit is assigned to a context vector $\vec{v}_c$ where initially all the elements of the vector are set to 0. Context vectors are then updated incrementally by the accumulation of the index vector $\vec{r}_i$ of the encountered context elements. The result is a VSM at reduced dimensionality that can be used to estimate the ℓ_1 distances between text units in the model.

Using the sample median, for given vectors $\vec{v}$ and $\vec{u}$, the approximate ℓ_1 distance between vectors, which we denote by $\hat{L}_1$, can be estimated by $\hat{L}_1(\vec{u},\vec{v}) = median\{|v_i - u_i|, i = 1,2,\cdots,m\}$, where m is the dimension of the VSM constructed by RMI, and $|.|$ denotes the modulus. Alternatively, as suggested above, $\hat{L}_1$ can be computed using the geometric mean, i.e. $\hat{L}_1 = (\prod_{i=1}^{m}|u_i - v_i|)^{1/m}$. In order to avoid overflow during the calculation of geometric mean, we use the arithmetic mean of logarithm-transformed values of $|u_i - v_i|$:

$$\hat{L}_1(\vec{u},\vec{v}) = \exp\big(\frac{1}{m}\sum_{i=1}^{m}\ln(|u_i - v_i|)\big). \quad (6)$$

In order to employ RMI for the construction of a VSM at reduced dimension, two model parameters should be decided: (a) the dimension of the VSM, which is shown by m, and (b) the number of non-zero elements in index vectors, which is determined by s in Equation 5. In contrast to the classic *one-dimension-per-context-element* methods of VSM construction,[5] the value of m in RPs and thus in RMI is chosen independently of the number of context elements n in the model. In RMI, as shown in [18], m is established by the probability and the maximum expected amount of distortions ϵ in pairwise distances and the number of vectors p in the model: a larger m yields to lower bounds on the distortion with a higher probability, i.e. $m = \mathrm{O}(\frac{\log(p)}{\epsilon^2})$. While a small m is desirable from the computational complexity outlook, the choice of m is often a trade-off between accuracy and efficiency. According to our experiment, $m > 400$ is suitable for most applications. The number of non-zero elements in index vectors, however, is decided by the number of context elements n and the sparseness of the VSM at its original dimension α. The proofs stated in [17] suggest $\mathrm{O}(\sqrt{\alpha n})$ as the value of s in Equation 5. In text-based applications, the sparsity of VSMs is considered to be around 0.01–0.0001. 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, it imposes higher computational complexity.

V. Evaluation and Experimental Results

The purpose of our reported evaluation is not to show the superiority of the ℓ_1 distance (thus RMI) to another similarity measure (e.g. the ℓ_2 distance, which is estimated in RI-constructed VSMs[6]) in a specific task. As a result, instead of a task-specific evaluation, we report the performance of RMI with respect to its ability to preserve the relative ℓ_1 distances between text units in a VSM. As stated earlier, the performance of the ℓ_1 distance for similarity measurement varies from one application to another, depending on the structure of the data that are being analyzed and the objective of the task in hand. We thus show that the relative ℓ_1 distance between a set of documents in a *document-by-term* model remains intact when using RMI with the suggested parameters.

In the designed experiment, a VSM is first constructed from the INEX-Wikipedia 2009 collection at its original high dimension. The corpus is a collection of 2,666,190 documents (articles) from a Wikipedia snapshot of October 2008 [19].[7] A pre-processing of the articles—i.e. white-space tokenization followed by the elimination of non-alphabetic tokens—results in a vocabulary of 2,533,854 terms. Each article in the dataset is represented by a high-dimensional vector; each dimension represents an entry in the obtained vocabulary. Hence, the constructed VSM using this one-dimension-per-context-element method has a dimensionality of 2.53 million. In order to keep the experiments tractable, we choose a list of 1000 random articles from the corpus. In the performed experiment, a document from the list is taken as the reference and using the constructed high-dimensional VSM, its ℓ_1 distance to the remaining 999 documents in the list is calculated. These documents are then sorted in ascending order by the calculated ℓ_1 distance to obtain a ranked list of documents. The process is repeated for all other documents in the list, which consequently gives 1000 lists of ranked documents.

The procedure described above is repeated to obtain the lists of ranked documents using the ℓ_1 distances that are computed in RMI-constructed VSMs. In these reiterations, the RMI's parameters, i.e. the dimension and the number of non-zero elements in index vectors, are set to different values. We expect the relative ℓ_1 distances between documents in RMI-constructed VSMs to be the same as in the original high-dimensional VSM. Hence, the obtained sorted lists of ranked documents from the RMI-constructed VSMs must be identical to the corresponding lists that are derived from the original high-dimensional VSM. Consequently, for each RMI-constructed VSM, the resulting sorted lists are compared with the obtained sorted lists from the original high-dimensional VSM using Spearman's rank correlation coefficient measure (ρ). We report the average of ρ over the 1000 lists of sorted documents ($\bar{\rho}$) to indicate the performance of RMI with respect to its ability in distance preservation: the closer $\bar{\rho}$ is to 1, the higher the performance of RMI.[8]

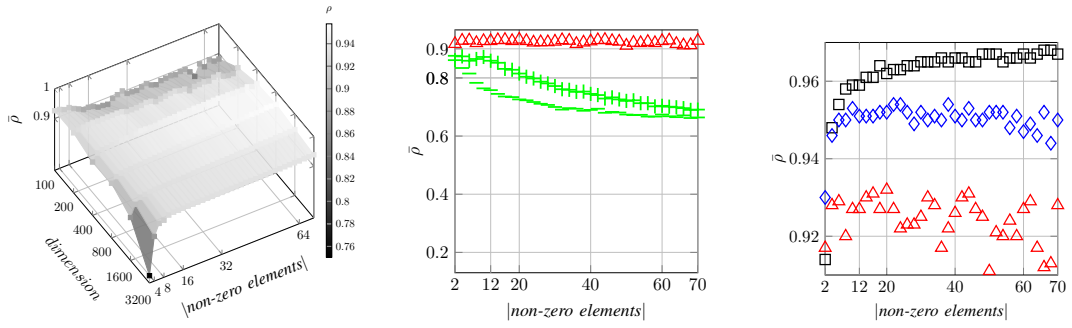
As shown in Figure 2a, when the dimension of the VSM is above 400 and the number of non-zero elements is more than 12, the relative distances obtained from the VSM constructed by RMI start to be analogous to the relative distances that are observed in the original VSM, i.e. a high correlation ($\bar{\rho} > 0.93$). As suggested in Section IV, when the dimension of the VSM increases, the probability of preserving distances increases. As a result, RMI at high dimension shows more stable performance than that at lower dimension. The models at lower dimension, however, converge faster than models of higher dimension; i.e. with less non-zero elements, VSMs of low dimension start to show a high correlation to the ℓ_1 distances in the original high-dimensional VSM. Figure 2b shows the same results presented in Figure 2a, however,

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

[6] Or, VSMs that are obtained after SVD truncation.

[7] The corpus can be obtained from http://goo.gl/E0mw7k.

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



(a) (b) (c)

Fig. 2. The ability of RMI to preserve the ℓ_1 distance is assessed by the observed average Spearman correlation $\bar{\rho}$ between the ranking of 1000 documents in the original high-dimensional VSM and RMI-construed VSMs at reduced dimensionality. Figure 2a shows the overall observed result when the dimension and the number of non-zero elements in index vectors (i.e. the RMI' parameters) are set differently. Figure 2b shows the same results only when the dimension of VSM is 200 (the red triangle markers). In this figure, the minimum value of $\bar{\rho}$-axis is set to the best observed correlation $\rho = 0.1375$ when distances are generated randomly (first baseline). The $+$ and $-$ marks (green marks) show $\bar{\rho}$ when ℓ_1 distance is estimated in RI-constructed VSMs of dimensionality 1600. Figure 2c, the same observed results are plotted only for RMI-constructed VSMs at the dimensionality of 200 (red triangles), 400 (blue diamonds) and 800 (black squares). It can be verified that an increase in the dimension of VSM results in an increase in $\bar{\rho}$.

only when the dimension of RMI-constructed VSM is 200. In this figure, we show two baselines. To generate the first baseline, the selected 1000 documents in the experiment are assigned to randomly generated distances and then sorted and compared by the calculated distances in the original high-dimensional VSM. This process is repeated 1000 times. We report the highest observed correlation of $\rho = 0.137$ as the first baseline. For these randomly generated distances, expectedly, the average correlation is almost zero, i.e. $\bar{\rho} = 0.00003$. For the second baseline, in order to support the earlier claim that RI-constructed VSMs do not preserve the ℓ_1 distances, we use RI to construct VSMs at the reduced dimension of 1600 and for various numbers of non-zero elements. In the RI-constructed VSMs, the ℓ_1 distances are then computed using the standard $\sum_{i=1}^{n} |v_i - u_i|$ as well as Equation 6 ('+' and '−' marks in Figure 2b, respectively). The observed $\bar{\rho}$ when documents are sorted using these calculated distances are shown as the second baseline. Figure 2c shows the presented $\bar{\rho}$ in Figure 2a only for VSMs of reduced dimensionality 200, 400 and 800.

VI. Conclusion

We introduced the RMI method, a novel technique for the construction of ℓ_1 normed VSMs at reduced dimensionality. RMI merges the construction of a VSM with the dimensionality reduction process. Hence, it creates a VSM directly at reduced dimension. RMI, therefore, alleviates the curse of dimensionality when similarity between text units is measured in ℓ_1 normed spaces. The RMI technique employs Cauchy random projections and a non-linear estimator to attain its goal. It can be seen as the peer of RI (used to measure similarity using the ℓ_2 distance) and TopSig (used to measure similarity using the Hamming distance), however, when similarity between text units is measured using the ℓ_1 distance. We further validated the ability of RMI-constructed VSMs to preserve the ℓ_1 distances in an experiment.

Acknowledgment

We would like to express our gratitude to the the anonymous reviewers. This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant Number SFI/12/RC/2289.

References

[1] P. D. Turney and P. Pantel, "From frequency to meaning: vector space models of semantics," *J. Artif. Int. Res.*, vol. 37, no. 1, 2010.

[2] P. Kanerva, J. Kristoferson, and A. Holst, "Random indexing of text samples for latent semantic analysis," in *Proceedings of CogSci*. 2000.

[3] M. Sahlgren, "An introduction to random indexing," in *Methods and Applications of Semantic Indexing Workshop at TKE 2005*, 2005.

[4] S. Geva and C. M. De Vries, "Topsig: Topology preserving document signatures," in *Proceedings of the CIKM '11*. ACM, 2011.

[5] C. M. De Vries and S. Geva, "Pairwise similarity of topsig document signatures," in *Proceedings of ADCS '12*. ACM, 2012, pp. 128–134.

[6] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, 1975.

[7] N. Kwak, "Principal component analysis based on l1-norm maximization," *TPAMI, IEEE Transactions on*, vol. 30, no. 9, 2008.

[8] J. Weeds, J. Dowdall, G. Schneider, B. Keller, and D. Weir, "Using distributional similarity to organise biomedical terminology," *Terminology*, vol. 11, no. 1, pp. 3–4, 2005.

[9] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *J. Assoc. Inf. Sci. Technol.*, vol. 41, no. 6, pp. 391–407, 1990.

[10] T. Cohen, R. Schvaneveldt, and D. Widdows, "Reflective Random Indexing and indirect inference: A scalable method for discovery of implicit connections," *J. Biomed. Inform.*, vol. 43, no. 2, 2010.

[11] D. Achlioptas, "Database-friendly random projections," in *Proceedings of PODS '01*. ACM, 2001, pp. 274–281.

[12] P. Li, T. J. Hastie, and K. W. Church, "Very sparse random projections," in *Proceedings of KDD '06*. ACM, 2006, pp. 287–296.

[13] W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," in *Contemporary Mathematics*. American Mathematical Society, 1984, vol. 26, pp. 189–206.

[14] S. Dasgupta and A. Gupta, "An elementary proof of a theorem of Johnson and Lindenstrauss," *Random Structures & Algorithms*, vol. 22, no. 1, pp. 60–65, 2003.

[15] B. Brinkman and M. Charikar, "On the impossibility of dimension reduction in l1," *J. ACM*, vol. 52, no. 5, pp. 766–788, Sep. 2005.

[16] P. Indyk, "Stable distributions, pseudorandom generators, embeddings and data stream computation," in *Proceedings of FOCS*. 2000.

[17] P. Li, "Very sparse stable random projections for dimension reduction in l_α $(0 < \alpha < 2)$ norm," in *Proceedings of KDD '07*, ACM, 2007.

[18] P. Li, T. J. Hastie, and K. W. Church, "Nonlinear estimators and tail bounds for dimension reduction in L_1 using Cauchy random projections," *J. Mach. Learn. Res.*, vol. 8, pp. 2497–2532, Dec. 2007.

[19] R. Schenkel, F. M. Suchanek, and G. Kasneci, "YAWN: A semantically annotated Wikipedia XML corpus," in *BTW*, vol. 103. 2007.