Combining Text Vector Representations for Information Retrieval

Maya Carrillo[1,2], Chris Eliasmith[3], and A. López-López[1]

[1] Coordinación de Ciencias Computacionales, INAOE,
Luis Enrique Erro 1, Sta.Ma. Tonantzintla, 72840, Puebla, Mexico
[2] Facultad de Ciencias de la Computación, BUAP,
Av. San Claudio y 14 Sur Ciudad Universitaria, 72570 Puebla, Mexico
{cmaya,allopez}@inaoep.mx
[3] Department of Philosophy, Department of Systems Design Engineering,
Centre for Theoretical Neuroscience, University of Waterloo,
200 University Avenue West Waterloo, Canada
celiasmith@uwaterloo.ca

Abstract. This paper suggests a novel representation for documents that is intended to improve precision. This representation is generated by combining two central techniques: Random Indexing; and Holographic Reduced Representations (HRRs). Random indexing uses co-occurrence information among words to generate semantic context vectors that are the sum of randomly generated term identity vectors. HRRs are used to encode textual structure which can directly capture relations between words (e.g., compound terms, subject-verb, and verb-object). By using the random vectors to capture semantic information, and then employing HRRs to capture structural relations extracted from the text, document vectors are generated by summing all such representations in a document. In this paper, we show that these representations can be successfully used in information retrieval, can effectively incorporate relations, and can reduce the dimensionality of the traditional vector space model (VSM). The results of our experiments show that, when a representation that uses random index vectors is combined with different contexts, such as document occurrence representation (DOR), term co-occurrence representation (TCOR) and HRRs, the VSM representation is outperformed when employed in information retrieval tasks.

1 Introduction

The vector space model (VSM) [1] for document representation supporting search is probably the most well-known IR model. The VSM assumes that term vectors are pair-wise orthogonal. This assumption is very restrictive because words are not independent. There have been various attempts to build representations for documents and queries that are semantically richer than only vectors based on the frequency of terms occurrence. One example is Latent Semantic Indexing (LSI), a word space model, which assumes that there is some underlying latent semantic structure (concepts) that can be estimated by statistical techniques. The traditional word space models produce a high dimensional vector space storing co-occurrence data in a matrix M known as co-occurrence matrix, where each row M_w represents a word and each column M_c

V. Matoušek and P. Mautner (Eds.): TSD 2009, LNAI 5729, pp. 24–31, 2009.

a context (a document or other word). The entry M_{wc} records the co-occurrence of word w in the context c. The M_w rows are vectors, whose size depends on the number of contexts, and are known as 'context vectors' of the words because they represent the contexts in which each word appears. Thus, an algorithm that implements a word space model has to handle the potentially high dimensionality of the context vectors, to avoid affecting its scalability and efficiency. Notably, the majority of the entries in the co-occurrence matrix will be zero given that most words occur in limited contexts.

The problems of very high dimensionality and data sparseness have been approached using dimension reduction techniques such as singular value decomposition (SVD). However, these techniques are computationally expensive in terms of memory and processing time. As an alternative, there is a word space model called Random Indexing [4], which presents an efficient, scalable, and incremental method for building context vectors. Here we explore the use of Random Indexing to produce context vector using document occurrence representation (DOR), and term co-occurrence representation (TCOR). Both DOR and TCOR can be used to represent the content of a document as a bag of concepts (BoC), which is a recent representation scheme based on the perception that the meaning of a document can be considered as the union of the meanings of its terms. This is accomplished by generating term context vectors from each term within the document, and generating a document vector as the weighted sum of the term context vectors contained within that document [4].

In DOR, the meaning of a term is considered as the sum of contexts in which it occurs. In this case, contexts are defined as entire documents. In TCOR the meaning of a term t is viewed as the sum of terms with which it co-occurs, given a window centered in t.

In addition to random indexing, we explore the use of linguistic structures (e.g., compound terms as: *operating system, information retrieval*; and binary relations as: subject-verb and verb-object) to index and retrieve documents. The traditional methods that include compound terms first extract them and then subsequently include these compound terms as new VSM terms. We explore a different representation of such structures, which uses a special kind of vector binding (called holographic reduced representations (HRRs) [3]) to reflect text structure and distribute syntactic information across the document representation. This representation has the benefit, over adding new terms, of preserving semantic relations between compounds and their constituents (but only between compounds to the extent that both constituents are similar). In other words, HRRs do not treat compounds as semantically independent of their constituents. A processing text task where HRRs have been used together with Random Indexing is text classification, where they have shown improvement under certain circumstances, using BoC as baseline [2].

The remainder of this paper is organized as follows. In Section 2 we briefly review Random Indexing. Section 3 introduces the concept of Holographic Reduced Representations (HRRs). Section 4 presents how to use HRRs to add information displaying text structure to document representations. Section 5 explains how different document representations were combined, aiming to improve precision. Section 6 describes the experiments performed. Section 7 shows the results that were obtained in experimental collections. Finally, Section 8 concludes the paper and gives some directions for further work.

2 Random Indexing

Random Indexing (RI) [4] is a vector space methodology that accumulates context vectors for words based on co-occurrence data. First, a unique random representation known as index vector is assigned to each context (either document or word), consisting of a vector with a small number of non-zero elements, which are either +1 or -1, with equal amounts of both. For example, if the index vectors have twenty non-zero elements in a 1024-dimensional vector space, they have ten +1s and ten -1s. Index vectors serve as indices or labels for words or documents. Second, index vectors are used to produce context vectors by scanning through the text and every time a target word occurs in a context, the index vector of the context is added to the context vector of the target word. Thus, with each appearance of the target word t with a context c the context vector of t is updated as follows:

$$ct+ = ic \tag{1}$$

where ct is the context vector of t and ic is the index vector of c. In this way, the context vector of a word keeps track of the contexts in which it occurred.

3 Holographic Reduced Representation

Two types of representation exist in connectionist models: localist, which uses particular units to represent each concept (objects, words, relationships, features); and distributed, in which each unit is part of the representation of several concepts. HRRs are a distributed representation and have the additional advantage that they allow the expression of structure using a circular convolution operator to bind terms (without increasing vector dimensionality). The circular convolution operator ($\otimes$) binds two vectors $\overrightarrow{x} = (x_0, x_1, ..., x_{n-1})$ and $\overrightarrow{y} = (y_0, y_1, ..., y_{n-1})$ to produce $\overrightarrow{z} = (z_0, z_1, ..., z_{n-1})$ where $\overrightarrow{z} = \overrightarrow{x} \otimes \overrightarrow{y}$ is defined as:

$$z_i = \sum_{k=0}^{n-1} x_k y_{i-k} \qquad i = 0 \text{ to } n-1 \text{ (subscripts are module-}n\text{)} \tag{2}$$

A finite-dimensional vector space over the real numbers with circular convolution and the usual definition of scalar multiplication and vector addition form a commutative linear algebra system, so all the rules that apply to scalar algebra also apply to this algebra [3]. We use this operator to combine words and represent compound terms and binary relations.

4 HRR Document Representation

We adopt HRRs to build a text representation scheme in which the document syntax can be captured and can help improve retrieval effectiveness. To define an HRR document representation, the following steps are done: a) we determine the index vectors for the vocabulary by adopting the random indexing method, described earlier; b) all documents are indexed adding the index vectors of the single terms they contain (IVR);

c) for each textual relation in a document, the index vectors of the involved words are bound to their role identifier vectors (using HRRs); d) The tf.idf-weighted sum of the resulting vectors is taken to obtain a single HRR vector representing the textual relation; e) HRRs of the textual relations, multiplied by an attenuating factor α, are added to the document vector (formed with the addition of the single term vectors), to obtain a single HRR vector representing the document, which is then normalized.

For example, given a compound term: *R = information retrieval*. This will be represented using the index vectors of its terms *information* ($\overrightarrow{r}_1$) and *retrieval* ($\overrightarrow{r}_2$), as each of them plays a different role in this structure (right noun/left noun). To encode these roles, two special vectors (HRRs) are needed: $\overrightarrow{role}_1, \overrightarrow{role}_2$. Then, the *information retrieval* vector is:

$$\overrightarrow{R} = (\overrightarrow{role}_1 \otimes \overrightarrow{r}_1 + \overrightarrow{role}_2 \otimes \overrightarrow{r}_2) \tag{3}$$

Thus, given a document D, with terms $t_1, t_2, \ldots, t_{x1}, t_{y1}, \ldots, t_{x2}, t_{y2}, \ldots, t_n$, and relations R_1, R_2 among terms $t_{x1}, t_{y1}; t_{x2}, t_{y2}$, respectively, its vector is built as:

$$\overrightarrow{D} = \langle \overrightarrow{t}_1 + \overrightarrow{t}_2 + \ldots + \overrightarrow{t}_n + \alpha((\overrightarrow{role}_1 \otimes \overrightarrow{t}_{x1} + \overrightarrow{role}_2 \otimes \overrightarrow{t}_{y1}) + (\overrightarrow{role}_1 \otimes \overrightarrow{t}_{x2} + \overrightarrow{role}_2 \otimes \overrightarrow{t}_{y2}))\rangle \tag{4}$$

where $\langle\ \rangle$ denotes a normalized vector and α is a factor less than one intended to lower the impact of the coded relations. Queries are represented in the same way.

5 Combining Representations

We explored several representations: index vector representation (IVR), which uses index vectors as context vectors, DOR, TCOR with a one-word window (TCOR1), and TCOR with a ten-word window (TCOR10). These four document representations were created using BoC. We then combined the similarities obtained from the different representations to check if they took into account different aspects that can improve precision. This combination involves adding the similarity values of each representation and re-ranking the list. Thus, IVR-DOR is created by adding the IVR similarity values to their corresponding values from DOR and re-ranking the list, where documents are now ranked according to the relevance aspects conveyed by both IVR and DOR. We create IVR-TCOR1 using the same process as described above, but now with the similarity lists generated by IVR and TCOR1. Finally, the two similarity lists IVR and TCOR10 are added to form IVR-TCOR10.

In addition, the similarity list obtained with HRR document representations, denoted as IVR+PHR, is also combined with DOR, TCOR1 and TCOR10 similarity lists to produce the IVR+PHR-DOR, IVR+PHR-TCOR1 and IVR+PHR-TCOR10 similarity lists, respectively. These combinations are performed to include varied context information. The following section outlines the experiments performed.

6 Experiments

The proposed document representation was applied to two collections: CACM, with 3,204 documents and 64 queries and NPL, with 11,429 documents and 93 queries. The

traditional vector space model (VSM) was used as a baseline, implemented using tf.idf weighting scheme and cosine function to determine vector similarity. We compared this against our representations, which used random indexing, the cosine as a similarity measure, and the same weighting scheme. We carried out preliminary experiments intended to assess the effects of dimensionality, limited vocabulary, and context definition; the following experiments were done using vectors of 4,096 dimensionality, removing stop words, and doing stemming, in the same way as for VSM. The experimental setup is described in the following sections.

6.1 First Set of Experiments: Only Single Terms

CACM and NPL collections were indexed using RI. The number of unique index vectors generated for the former was 6,846 (i.e. terms) and 7,744 for the latter. These index vectors were used to generate context vectors using DOR, TCOR1 and TCOR10. We consider four experiments: a) IVR b) IVR-DOR c) IVR-TCOR1 d) IVR-TCOR10 as described in section 5. It is worth mentioning that the results independently obtained with DOR and TCOR alone were below VSM precision by more than 20%.

6.2 Second Set of Experiment: Noun Phrases

Compound terms were extracted after parsing the documents with Link Grammar [5], doing stemming, and selecting only those consisting of pairs of collocated words. The compound terms obtained for CACM were 9,373 and 18,643 for NPL. These compound terms were added as new terms to the VSM (VSM+PHR). The experiments performed for comparison to this baseline were: a) IVR+PHR, which represents documents as explained in section 4, using the term index vectors, and HRRs to encode compound terms, taking α equal to 1/6 in (4). b) IVR+PHR-DOR, c) IVR+PHR-TCOR1, and d) IVR+PHR-TCOR10, as described in section 5.

6.3 Third Set of Experiments: Binary Relations

The relations to be extracted and included in this vector representation were: compound terms (PHR), verb-object (VO) and subject-verb (SV). These relationships were extracted from the queries of the two collections using Link Grammar and MontyLingua 2.1 [6]. The implementation of the Porter Stemmer used in the experiments came from the Natural Language Toolkit 0.9.2. In this experiment, all stop words were eliminated and stemming was applied to all the relations. If one of the elements of composed terms or SV relations had more than one word, only the last word was taken. The same criterion was applied for the verb in the VO relation; the object was built only with the first set of words extracted and the last word taken, but only if the first word of the set was neither a preposition nor a connective.

Afterwards, a similarity file using only simple terms was generated (IVR). Following this, the HRRs for PHR relations were built for documents and queries and another similarity file was defined. This process was repeated to generate two additional similarity files, but now using SV and VO relations. Then, three similarity files for the extracted relations were built. The IVR similarity file was then added to the PHR similarity, multiplied by a constant of less than one, and the documents were sorted again according

to their new value. Afterwards, the SV and VO similarity files were added and the documents once again sorted. Therefore, the similarity between a document d and a query q is calculated with (5), where β, δ, γ are factors less than 1.

$$similarity(q,d) = IVRsimilarity(q,d) + \beta\ PHRsimilarity(q,d) + \delta\ SVsimilarity(q,d) + \gamma\ VOsimilarity(q,d) \quad (5)$$

7 Results

In Tables 1 and 2, we present the calculated mean average precision (MAP - a measure to assess the changes in the ranking of relevant documents), for all our experiments. IVR when considering single terms or compound terms with TCOR reaches higher MAP values than VSM in all cases. For NPL collection, IVR combined with DOR also surpasses the VSM MAP; even the MAP for IVR+PHR is higher than the MAP obtained for VSM+PHR. For CACM, the results obtained with IVR-TCOR10 were found to be statistically significant in a 93.12% confidence interval. For NPL, however, the results for IVR-TCOR10 were significant in a 99.8% confidence interval. IVR+PHR-TCOR1 was significant in a 97.8% confidence interval, and finally IVR+PHR-DOR and IVR+PHR-TCOR10 were significant in a 99.99% confidence interval.

Finally, the experimentation using binary relations was done after extracting the relations for the queries of each collection. Table 3 shows the number of queries for

Table 1. MAP comparing VSM against IVR and IVR-DOR

	Single terms				
	VSM	IVR	%of change	IVR-DOR	% of change
CACM	0.2655	0.2541	-4.28	0.2634	-0.81
NPL	0.2009	0.1994	-0.76	0.2291	**14.02**
	Terms including compound terms				
	VSM+PHR	IVR+PHR	% of change	IVR+ PHR-DOR	% of change
CACM	0.2715	0.2538	-6.54	0.2631	-3.10
NPL	0.1857	0.1988	**7.05**	0.2291	**23.40**

Table 2. MAP comparing VSM against IVR-TCOR1 and IVR-TCOR10

	Single terms				
	VSM	IVR-TCOR1	% of change	IVR-TCOR10	% of change
CACM	0.2655	0.2754	**3.72**	0.3006	**13.22**
NPL	0.2009	0.2090	**4.01**	0.2240	**11.48**
	Terms including compound terms				
	VSM+ PHR	IVR+ PHR TCOR1	% of change	IVR+ PHR TCOR10	% of change
CACM	0.2715	0.2743	**1.04**	0.3001	**10.54**
NPL	0.1857	0.2088	**12.43**	0.2232	**20.22**

Table 3. Number of queries with selected relations per collection

Collection	Compound terms	Subject-Verb	Object-Verb
CACM	48	28	33
NPL	66	1	3

Table 4. MAP comparing the VSM with IVR after adding all the relations

VSM	IVR	% of change	IVR+ PHR	% of change
0.2563	0.2570	**0.28**	0.2582	**0.74**
	IVR+PHR+SV	% of change	IVR+PHR+SV+VO	% of change
	0.2610	**1.82**	0.2693	**5.07**

each collection that had at least one relation of the type specified in the column. NPL queries had very few relations other than compound terms. Consequently, we only experimented using CACM. For this collection, we worked with 21 queries, which had all the specified relations. The value given to β in (5) was 1/16 and to δ and γ 1/32, determined by experiments.

The MAP reached by VSM and the proposed representation with the relations joined is shown in table 4, where the average percentage of change goes from 0.27% for IVR to 5.07% after adding all the relations.

8 Conclusion and Future Research

In this paper, we have presented a proposal for representing documents and queries using random indexing. The results show that this approach is feasible and is able to support the retrieval of information, while reducing the vector dimensionality when compared to the classical vector model. The document representation, using index vector generated by random indexing and the HRRs to encode textual relations, captures some syntactical details that improve precision, according to the experiments. The semantics expressed by contexts either using DOR or TCOR added to our representation also improves the retrieval effectiveness, seemingly by complementing the terms coded alone, something that, as far as we know, has not been experimented on before.

The representation can also support the expression of other relations between terms (e.g. terms forming a named entity). We are in the process of further validating the methods in bigger collections, but we require collections with sufficient features (i.e. queries with binary relations) to fully assess the advantages of our model.

Acknowledgements

The first author was supported by scholarship 217251/208265 granted by CONACYT, while the third author was partially supported by SNI, Mexico.

References

1. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM 18(11), 613–620 (1975)
2. Fishbein, J.M., Eliasmith, C.: Integrating structure and meaning: A new method for encoding structure for text classification. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 514–521. Springer, Heidelberg (2008)
3. Plate, T.A.: Holographic Reduced Representation: Distributed representation for cognitive structures. CSLI Publications (2003)
4. Sahlgren, M., Cöste, R.: Using Bag-of-Concepts to Improve the Performance of Support Vector Machines in Text Categorization. In: Procs. of the 20th International Conference on Computational Linguistics, pp. 487–493 (2004)
5. Grinberg, D., Lafferty, J., Sleator, D.: A Robust Parsing Algorithm for Link Grammars, Carnegie Mellon University, Computer Science Technical Report CMU-CS-95-125 (1995)
6. Liu, H.: MontyLingua: An end-to-end natural language processor with common sense. web.media.mit.edu/ hugo/montylingua (2004)