Natural Language Engineering: page 1 of 38. © Cambridge University Press 2011
doi:10.1017/S1351324911000271

*Evaluating vector space models with canonical correlation analysis**

SAMI VIRPIOJA[1], MARI-SANNA PAUKKERI[1], ABHISHEK TRIPATHI[2], TIINA LINDH-KNUUTILA[1], and KRISTA LAGUS[1]

[1]*Department of Information and Computer Science, Aalto University School of Science*
P.O. Box 15400, FI-00076 Aalto, Finland
e-mails: sami.virpioja@tkk.fi, mari-sanna.paukkeri@tkk.fi, tiina.lindh-knuutila@tkk.fi, krista.lagus@tkk.fi

[2]*Department of Computer Science, University of Helsinki, Finland*
and
Xerox Research Centre Europe (XRCE)
6, Chemin de Maupertuis, 38240, Meylan, France
e-mail: abhishektripathi.at@gmail.com

(*Received 11 October 2010; revised 14 July 2011; accepted 31 July 2011*)

Abstract

Vector space models are used in language processing applications for calculating semantic similarities of words or documents. The vector spaces are generated with feature extraction methods for text data. However, evaluation of the feature extraction methods may be difficult. Indirect evaluation in an application is often time-consuming and the results may not generalize to other applications, whereas direct evaluations that measure the amount of captured semantic information usually require human evaluators or annotated data sets. We propose a novel direct evaluation method based on canonical correlation analysis (CCA), the classical method for finding linear relationship between two data sets. In our setting, the two sets are parallel text documents in two languages. A good feature extraction method should provide representations that reflect the semantic content of the documents. Assuming that the underlying semantic content is independent of the language, we can study feature extraction methods that capture the content best by measuring dependence between the representations of a document and its translation. In the case of CCA, the applied measure of dependence is correlation. The evaluation method is based on unsupervised learning, it is language- and domain-independent, and it does not require additional resources besides a parallel corpus. In this paper, we demonstrate the evaluation method on a sentence-aligned parallel corpus. The method is validated by showing that the obtained results with bag-of-words representations are intuitive and agree well with the previous findings. Moreover, we examine

* We are grateful to the anonymous reviewers for their detailed and insightful comments on this paper. We also thank our colleagues Marcus Dobrinkat, Timo Honkela, Arto Klami, Oskar Kohonen, and Jaakko Väyrynen for their feedback and advice. SV, MP, TL, and KL belong to the Adaptive Informatics Research Centre, an Academy of Finland Centre of Excellence. AT was at Helsinki Institute for Information Technology HIIT and Department of Computer Science, University of Helsinki when this work was done. SV was supported by Graduate School of Language Technology in Finland, MP by Finnish Graduate School in Language Studies, and KL was supported by Academy of Finland (decision number 218214).

the performance of the proposed evaluation method with indirect evaluation methods in simple sentence matching tasks, and a quantitative manual evaluation of word translations. The results of the proposed method correlate well with the results of the indirect and manual evaluations.

1 Introduction

In many language processing tasks, textual data are transformed into vectorial form for efficient computation of similarities of words or documents. In the information retrieval (IR) community (Salton, Wong and Yang 1975), these are called vector space models. Other applications for vector space models include, for instance, word sense disambiguation (Schütze 1992), text categorization (Lewis 1992), cross-document coreferencing (Bagga and Baldwin 1998), and bilingual lexicon acquisition (Sahlgren and Karlgren 2005).

One of the main challenges in vector space models' research is the evaluation of the feature extraction methods that are used for constructing vector representations. The methods include feature selection, feature weighting, dimensionality reduction, and normalization. Even if the target application is known, *indirect evaluation* in the application setting is rather time-consuming, which makes it difficult to test many different parameters of feature extraction. A method for quick estimation of the quality of the produced representations would allow to compare a large number of parameters, and to select only the best ones for the application evaluation. In addition, if the application consists of several components, it would be beneficial to be able to measure the performance of each component separately. However, as pointed out by Sahlgren (2006a), simple and robust approaches for *direct evaluation* of vector representations are missing.

In this paper, we propose a direct evaluation method for vector space models of documents. The method is based on canonical correlation analysis (CCA). CCA has been applied to infer semantic representation between multimodal sources (Vinokourov, Shawe-Taylor and Cristianini 2003; Hardoon, Szedmak and Shawe-Taylor 2004). The parallel documents in two languages can be seen as two views of the same underlying semantics (Mihalcea and Simard 2005). If the evaluated feature extraction methods captured the language-independent semantic intention that is common to the aligned documents, then the produced features should have a high dependence. CCA finds the maximally dependent subspaces for the two sets of features using correlation as the measure of dependence, thus providing an efficient means of evaluating the feature extraction methods.

We demonstrate the proposed evaluation method by comparing various vector space models for sentences. There are several reasons for using sentences rather than words or documents. Word representations cannot be evaluated as such, because there is no one-to-one correspondence of words in different languages, and CCA needs a mapping of samples (words) between two data sets. In contrast, sentences are less ambiguous in meaning, and the assumption of shared semantic intention is reasonable. In addition, sentences are used as basic units in many natural language

processing applications, such as machine translation and question answering. A practical benefit is that large multilingual sentence-aligned corpora are readily available.

Although our experiments concentrate on evaluation of sentence representations, the proposed evaluation method is useful for many applications that utilize vector space models, given that there is an aligned corpus available. Unlike evaluations based on human language tests, our method is unsupervised and language-independent, and can be used with various large data sets. Moreover, it is faster and provides a more general measure of quality than indirect evaluation in applications.

The rest of the paper is organized as follows. Section 2 explains canonical correlation analysis and reviews earlier work that applies it to language data. Section 3 describes the vector space models of language and reviews earlier work related to the evaluation of vector space models. The proposed evaluation method is explained in Section 4, together with some examples on artificial data sets. In Section 5, the feasibility of the evaluation method is validated with real data from a sentence-aligned multilingual corpus. In Section 6, we discuss extensions and other possible uses for the evaluation framework. Finally, we conclude the work in Section 7.

2 Canonical correlation analysis

Canonical correlation analysis, originally proposed by Hotelling (1936), is a classical linear method for finding relationships between two sets of variables. It finds linear projections for each set of variables so that the correlation between the projections is maximized (Borga 1998; Bach and Jordan 2003; Hardoon *et al.* 2004).

Consider two column vectors of random variables $\mathbf{x} = [x_1, \ldots, x_{D_x}]^{\mathrm{T}}$ and $\mathbf{y} = [y_1, \ldots, y_{D_y}]^{\mathrm{T}}$ with zero means. For each variable pair, we want to find linear transformations into scalars, $u_1 = \mathbf{a}^{\mathrm{T}}\mathbf{x}$ and $v_1 = \mathbf{b}^{\mathrm{T}}\mathbf{y}$, so that the correlation between the scalars is maximized:

$$\rho_1 = \max_{\mathbf{a},\mathbf{b}} \mathrm{corr}(u_1, v_1) = \max_{\mathbf{a},\mathbf{b}} \frac{E[\mathbf{a}^{\mathrm{T}}\mathbf{x}\mathbf{y}^{\mathrm{T}}\mathbf{b}]}{\sqrt{E[\mathbf{a}^{\mathrm{T}}\mathbf{x}\mathbf{x}^{\mathrm{T}}\mathbf{a}]E[\mathbf{b}^{\mathrm{T}}\mathbf{y}\mathbf{y}^{\mathrm{T}}\mathbf{b}]}} \tag{1}$$

Correlation ρ_1 is the first canonical correlation and u_1 and v_1 are the first canonical variates. The subsequent canonical variates, u_i and v_i, are set to be maximally correlated as in (1) with the restriction that they are uncorrelated with all the previous variates, that is, $E[u_i u_j] = E[v_i v_j] = E[u_i v_j] = 0$ for all $i \neq j$. In total, there can be $D = \min(D_x, D_y)$ canonical variates and correlations.

2.1 Estimating canonical correlations

In practice, the expectations in (1) are replaced by sample-based estimates computed from observation matrices $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]$, resulting in a sample canonical correlation

$$\rho_1 = \max_{\mathbf{a},\mathbf{b}} \frac{\mathbf{a}^{\mathrm{T}}\mathbf{C}_{xy}\mathbf{b}}{\sqrt{\mathbf{a}^{\mathrm{T}}\mathbf{C}_{xx}\mathbf{a}}\sqrt{\mathbf{b}^{\mathrm{T}}\mathbf{C}_{yy}\mathbf{b}}} \tag{2}$$

Here, $\mathbf{C}_{xy} = \mathbf{C}_{yx}{}^{\mathrm{T}}$ is a between-sets covariance matrix and $\mathbf{C}_{xx}$ and $\mathbf{C}_{yy}$ are within-sets covariance matrices of the two random variables $\mathbf{x}$ and $\mathbf{y}$. Unbiased estimates

of the covariance matrices can be obtained by

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_{xx} & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & \mathbf{C}_{yy} \end{pmatrix} \approx \frac{1}{N-1} \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix} \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}^{\mathrm{T}} \tag{3}$$

where $\mathbf{C}$ is the full correlation matrix and N is the sample size.

Since the solution of (2) is not affected by the re-scaling of $\mathbf{a}$ or $\mathbf{b}$, the choice of re-scaling is arbitrary, and thus the maximization problem is equal to maximizing the numerator subject to

$$\mathbf{a}^{\mathrm{T}}\mathbf{C}_{xx}\mathbf{a} = \mathbf{b}^{\mathrm{T}}\mathbf{C}_{yy}\mathbf{b} = 1 \tag{4}$$

As shown by Bach and Jordan (2003), CCA reduces to solving the following generalized eigenvalue problem:

$$\begin{pmatrix} 0 & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} = \rho \begin{pmatrix} \mathbf{C}_{xx} & 0 \\ 0 & \mathbf{C}_{yy} \end{pmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} \tag{5}$$

which gives $D_x + D_y$ eigenvalues $\{\rho_1, -\rho_1, \ldots, \rho_D, -\rho_D, 0, \ldots, 0\}$, such that $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_D$. The eigenvectors $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_D]^{\mathrm{T}}$ and $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_D]^{\mathrm{T}}$ corresponding to D non-zero canonical correlations are the basis vectors for the canonical variates $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_D]^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}}\mathbf{X}$ and $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_D]^{\mathrm{T}} = \mathbf{B}^{\mathrm{T}}\mathbf{Y}$. Furthermore, the canonical variates are orthogonal ($\mathbf{U}\mathbf{U}^{\mathrm{T}} = \mathbf{V}\mathbf{V}^{\mathrm{T}} = \mathbf{I}$).

The estimates of canonical correlations depend heavily on the sample size and the dimensionality of random variables. A standard condition in classical CCA is $N/(D_x + D_y) \gg 1$. If the ratio is small, the sample covariance matrix $\mathbf{C}_{xy}$ may become ill-conditioned. It leads to a trivial or over-fitted CCA solution with canonical correlation of exactly one. Furthermore, the sample covariance matrices $\mathbf{C}_{xx}$ and $\mathbf{C}_{yy}$ may also be singular or near singular, leading to unreliable estimates of their inverses. One way to solve this issue is to introduce some kind of regularization (Leurgans, Moyeed and Silverman 1993; De Bie and De Moor 2003; Hardoon *et al.* 2004) by introducing smoothing to modify the constraints in (4). The regularized variant is solved through the same optimization problem, but a small positive value is added to the diagonal of $\mathbf{C}_{xx}$ and $\mathbf{C}_{yy}$ in the eigenvalue problem.

2.2 Canonical factor loadings

Canonical correlation analysis can be interpreted in terms of canonical factor loadings. In factor analysis, a loading is defined as a simple correlation between a variable and a factor. The square of the loading gives the variance of the variable explained by the factor (Harman 1960; Rummel 1970). Canonical factor loadings can be analogously defined as correlations between the original variable ($\mathbf{x}_j$ or $\mathbf{y}_j$) and each canonical variate ($\mathbf{u}_i$ or $\mathbf{v}_i$) in both the data sets:

$$l_{x(ij)} = \mathrm{corr}(\mathbf{u}_i, \mathbf{x}_j) \qquad l_{y(ij)} = \mathrm{corr}(\mathbf{v}_i, \mathbf{y}_j) \tag{6}$$

The loadings measure which variable is involved in which canonical variate and to what extent. Hence, a variable with a large canonical factor loading should be given more weight while deriving the interpretation of respective canonical variate. Moreover, the sum of the squared factor loadings divided by the number of variables

in the set is the proportion of variance in the set explained by the given canonical variate. In Section 5.4, we use the factor loadings for manual inspection of the variates.

2.3 CCA's connection to mutual information

There is a simple relationship between canonical correlation and mutual information (MI) for Gaussian random variables (Bach and Jordan 2003). Given two Gaussian random variables $\mathbf{x} = [x_1, \ldots, x_{D_x}]^{\mathrm{T}}$ and $\mathbf{y} = [y_1, \ldots, y_{D_y}]^{\mathrm{T}}$, the MI, $I(\mathbf{x}; \mathbf{y})$ can be written as

$$I(\mathbf{x}; \mathbf{y}) = -\frac{1}{2} \ln \left(\frac{|\mathbf{C}|}{|\mathbf{C}_{xx}||\mathbf{C}_{yy}|} \right) \tag{7}$$

where $|.|$ denotes the determinant of a matrix.

If $\mathbf{C}_{xx}$ and $\mathbf{C}_{yy}$ are invertible, the product of the eigenvalues is equal to the ratio of determinants in (7). Consequently, MI can be written in terms of canonical correlations (Kay 1992):

$$I(\mathbf{x}; \mathbf{y}) = -\frac{1}{2} \sum_{i=1}^{D} \ln(1 - \rho_i{}^2) \tag{8}$$

2.4 Nonlinear extensions to CCA

In many applications, it may not be sufficient to find linear dependence. One way to capture nonlinear dependence using CCA is to allow nonlinear transformations. Several authors (Lai and Fyfe 2000; Akaho 2001; Melzer, Reiter and Bischof 2001) have presented a CCA extension that enables nonlinear transformations using kernel functions. Kernel canonical correlation analysis (KCCA) has been further studied, for instance, by Bach and Jordan (2003) and Hardoon *et al.* (2004). In KCCA, the correlation matrices are replaced with a kernel function in the dual form. The data are projected into a feature space of high dimensionality H using kernel functions $\mathcal{K}$,

$$\mathcal{K} : \mathbb{R}^{D \times N} \mapsto \mathbb{R}^{H \times N}, \quad D < H \tag{9}$$

before computing CCA in the kernel space. Due to the higher dimensionality in the kernel space, KCCA overfits badly. In consequence, proper regularization is crucial for non-trivial learning (Bach and Jordan 2003; Hardoon *et al.* 2004).

2.5 Applying CCA to language data

Canonical correlation analysis and its variations have already been utilized in many applications of natural language processing. Cross-language IR is one of the applications in which CCA has been applied to a bilingual corpus. Mate retrieval is an IR task in which a document in a source language is used as the query and the corresponding document ('mate') in a target language is considered to be the only relevant document to the query. Vinokourov *et al.* (2003) use KCCA in mate

retrieval task for a sentence-aligned English–French corpus in which the documents are single paragraphs. In their experiments, KCCA performs significantly better than latent semantic indexing. The work is extended by Li and Shawe-Taylor (2007) by applying KCCA to a pair of languages from two language families, Japanese and English, with results that correspond to the results with the English–French corpus. Hardoon and Shawe-Taylor (2007) compare KCCA with linear kernel and a sparse CCA extension that uses sparsity constraints for the projection vectors in mate retrieval tasks for English–French and English–Spanish corpora. When there are many input features (words) and many enough projections, sparse CCA provides as good precision as KCCA, while the canonical variates are more interpretable because of their sparsity.

Another approach using CCA with bilingual corpus is the task to learn bilingual lexicons from two comparative monolingual corpora (Haghighi *et al.* 2008). Tripathi, Klami and Kaski (2008) independently propose a similar approach to infer matching of objects in two different views. Tripathi, Klami and Virpioja (2010) demonstrate it by matching sentences in two languages. They also extend the method for KCCA and obtain statistically significant improvements to the matching accuracy. Minier, Bodó and Csató (2007) apply KCCA to monolingual text categorization task, in which Wikipedia-based kernels are used to give word distributional representation for English documents. In their experiments, linear kernels perform better than nonlinear kernels.

3 Vector space models for language

Vector space models are a standard way to represent documents or words as vectors of features. The model provides a solution to the problem of representing symbolic information (words) in numerical form for computational processing (Salton 1971). In a vector space, similar items are close to each other and the closeness can be measured using vector similarity measures.

As an example of vector space models, a set of documents can be represented by the words they contain. Document j is represented by the vector $\hat{\mathbf{x}}_j$ containing the occurrences $c(i, j)$ of words, $i = 1, \ldots, M$ in the document:

$$\hat{\mathbf{x}}_j = [c(1, j), c(2, j), \ldots, c(M, j)]^{\mathrm{T}} \tag{10}$$

Matrix $\widehat{\mathbf{X}}$ that consists of the vector $\hat{\mathbf{x}}_j$ is called a word-document matrix. In these kind of representations, the word order information is discarded, and hence these are called bag-of-words representations (Schütze and Pedersen 1995). Different units of representation, such as index terms, letters, morphemes, or their sequences (n-grams, phrases), can be used as well.

Similarly as with the bag-of-words representation of documents, the words can be represented in terms of the documents in which they occur. Smaller contexts than documents, such as sentences or fixed-width windows, can also be used. The word-document matrix, or more generally, the feature-context matrix, contains the frequencies of the words in the contexts and thus represents first-order similarity (Rapp 2002). Second-order similarities can be observed by collecting a

word–word matrix, where the values are co-occurrences of words within some contexts, or a document–document matrix, where the values define how many common features are possessed by the documents. The second-order matrices can be obtained, for example, by computing $\widehat{\mathbf{X}}\widehat{\mathbf{X}}^{\mathrm{T}}$ for a word–word matrix or $\widehat{\mathbf{X}}^{\mathrm{T}}\widehat{\mathbf{X}}$ for a document–document matrix. A word–word matrix for short context windows often provides paradigmatic associations instead of syntagmatic associations that are obtained from first-order similarities (Rapp 2002).

3.1 Dimensionality reduction

The dimensionality of a feature-context matrix $\widehat{\mathbf{X}} \in \mathbb{R}^{M \times N}$ may be very high due to a large number of features M (e.g., words or index terms) or a large number of contexts N (e.g., documents, sentences, or neighboring words). To reduce the computational cost of calculating the similarities in the vector space, it is common to use dimensionality reduction. The methods for reducing the dimensionality can be divided into two families of approaches: feature selection and feature extraction (Schütze, Hull and Pedersen 1995; Sebastiani 2002; Alpaydin 2010). Sebastiani (2002) gives a comprehensive review on different feature selection and extraction methods for vector space models.

Feature selection. In feature selection, the task is to choose K dimensions out of the M original dimensions that give as much information as possible. The rest $M–K$ dimensions are discarded. Feature selection can be done systematically whenever it is possible to repeatedly evaluate the representation (Alpaydin 2010). In vector spaces constructed for language data, it is common to apply heuristic preprocessing, such as stemming, exclusion of too frequent and too rare words, or removal of non-alphabet characters, although more sophisticated methods have also been applied (Sebastiani 2002).

Feature extraction. In feature extraction, or reparameterization, the task is to find a new set of K dimensions that are combinations of the original M dimensions. That is, given a data set $\widehat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_N]$, where $\hat{\mathbf{x}}_i \in \mathbb{R}^M$, and a distance or similarity function $d(\cdot, \cdot)$, the task is to define a projection

$$\mathcal{R} : \mathbb{R}^{M \times N} \mapsto \mathbb{R}^{K \times N} \quad \text{s.t.} \quad d(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j) \approx d(\mathcal{R}(\hat{\mathbf{x}}_i), \mathcal{R}(\hat{\mathbf{x}}_j)) \tag{11}$$

Usually, this is accomplished by finding a linear projection $\mathbf{X} = \mathbf{R}\widehat{\mathbf{X}}$, where $\mathbf{R} \in \mathbb{R}^{K \times M}$ a projection matrix.

If the distance function $d(\cdot, \cdot)$ measures the Euclidean distance, the optimal linear solution for (11) can be found using the singular value decomposition (SVD) of $\widehat{\mathbf{X}}$: $\widehat{\mathbf{X}} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$, where the orthogonal matrices $\mathbf{U}$ and $\mathbf{V}$ contain the left and right singular vectors of $\widehat{\mathbf{X}}$ and the diagonal matrix $\mathbf{D}$ contains the respective singular values. The projection of $\widehat{\mathbf{X}}$ into the space spanned by the left singular vectors corresponding to the K largest singular values, $\mathbf{X} = \mathbf{U}_K\widehat{\mathbf{X}} = \mathbf{D}\mathbf{V}_K{}^{\mathrm{T}}$, gives the best mean square error solution.

A common application of SVD is to calculate principal component analysis (PCA), that is, the projection of $\widehat{\mathbf{X}}$ into the space spanned by the orthogonal components

of the largest variance. The use of SVD on text document data, dating back to Benzécri (1973), is often referred to as latent semantic analysis (LSA) (Deerwester *et al.* 1990). SVD, as well as probabilistic methods, such as probabilistic latent semantic analysis (PLSA) by Hofmann (1999) and latent dirichlet allocation (LDA) by Blei, Ng and Jordan (2003), exploit second-order statistics and generalize the data besides reducing the dimensionality. For instance, the latent space found for documents using LSA often combines the individual terms into more general topics. The methods can also address the problems of polysemy and synonymy (Deerwester *et al.* 1990).

A computationally light, but non-optimal way of reducing dimensionality is to project the data with random vectors that are nearly orthogonal. If the randomly selected subspace has a sufficiently high dimension, the distances between the data points are approximately preserved (Johnson and Lindenstrauss 1984). This approach has been addressed by several names: random projection (Ritter and Kohonen 1989), random mapping (Kaski 1998), and random indexing (Kanerva, Kristoferson and Holst 2000).

3.2 Weighting and normalization

Plain word document co-occurrence data give much weight to frequent words in the document collection. Different weighting schemes can be utilized to improve performance by giving weight to terms that represent best the semantic content . The schemes can be divided into global and local weighting schemes. Global weights indicate the overall importance of a term in the collection and are applied to each term in all the documents, whereas local weights are applied to each term in one document. The final weight is the product of the global and local weights. In the following, we describe the weighting schemes used in this paper. For a textbook description, cf., for example, Manning and Schütze (1999) or Manning, Raghavan and Schütze (2008).

Local weighting. The term frequency (tf) is an indicator of the saliency of a term, but often the effect of a raw count $c(i, j)$ is too large. Dampening of term frequency with a logarithm is common, yielding to logarithmic term frequency: $\log(1 + c(i, j))$. Further alternative is to use binary weights by simply discarding the term frequencies and using ones for all non-zero entries.

Global weighting. The global weighting schemes used in our experiments are summarized in Table 1. The most commonly used global weighting scheme, inverse of the document frequency (idf), assigns a high weight to terms that occur only in few documents and thus refer to very specific concepts. For term i, idf is the total number of documents N divided by the number of documents in which the word i occurs. In order to dampen the effect of the weight, usually logarithmic idf (log-idf) is applied (Jones 1972), but different functions, such as square root (sqrt-idf) and identity (lin-idf), can also be applied. Entropy weighting, based on information theoretic principles, assigns the minimum weight to terms for which the

Table 1. *Five global weightings of words. N is the number of documents in a document collection, $c(i,j)$ is the term frequency of word i in document j, $g(i) = \sum_j c(i,j)$ is the global frequency of word i in the whole collection, $d(i) = |\{j : c(i,j) > 0\}|$ is the document frequency for word i, and σ_i^2 is the sample variance for the term frequencies of word i*

Weighting	Coefficient for feature i
Logarithmic idf (log-idf)	$\log \frac{N}{d(i)}$
Square root idf (sqrt-idf)	$\sqrt{\frac{N}{d(i)}}$
Linear idf (lin-idf)	$\frac{N}{d(i)}$
Entropy weighting (entropy)	$1 - \sum_j \frac{p_{ij} \log p_{ij}}{\log N}$, where $p_{ij} = \frac{c(i,j)}{g(i)}$
Variance normalization (var1)	$\sigma_i^{-1} = \left(\frac{1}{N-1} \sum_j \left(c(i,j) - \frac{g(i)}{N}\right)^2\right)^{-\frac{1}{2}}$

distribution over documents is close to uniform and the maximum weight to terms that are concentrated in a few documents (Dumais 1991). Another method, more common for non-discrete data, is to normalize the variances of the features to one.

Length normalization. The length of the obtained vectors varies across the documents. The length depends on both the number of words present in each document and the applied local and global weightings. The similarities between the documents are often calculated with cosine similarity measure, which neglects the vector lengths (Salton and Buckley 1988). If some other distance measure is applied, the vectors can be explicitly normalized using, for example, L2 or L1 norms.

3.3 Evaluation methods for vector space models

The methods for measuring the quality of vector representations can be categorized as direct and indirect methods (Sahlgren 2006a). The direct methods compare the similarities in a vector space with external data, such as association norms or synonym tests, whereas the indirect methods measure the ability to solve a particular application task.

3.3.1 Indirect evaluation

Indirect methods have been used commonly for evaluating vector representations. The creation of a vector space has not been traditionally viewed as a research problem itself, but an intermediate phase in solving other natural language processing problems. The vector space has been applied to a task and the performance was evaluated for the task. Thus, the used evaluation methods may not generalize the applicability to other tasks.

In the IR community (see, e.g., Manning *et al.* 2008), the quality of vector representations is often measured using the IR results for evaluation. In document retrieval, for example, the evaluation is based on measuring how well the IR system

is able to rank documents according to the query. The list of correct documents for each query has been usually prepared manually.

Cross-language evaluations in IR are straightforward extensions to the monolingual IR when parallel corpora are available. If the best matching documents for the monolingual IR query are known, the corresponding (translation) documents in the second language are also known. There are multilingual test collections available, such as data from evaluation conferences TREC,[1] CLEF,[2] and NTCIR.[3]

Word sense disambiguation is another problem in which the vector space model can be utilized (Schütze 1992). Rather than mapping documents to a vector space, the word tokens in a corpus are mapped to the vector space and the different meanings of the same word type are disambiguated by clustering the word vectors. The evaluation of the vector space is conducted by a test set of words with two or more senses.

Word representations can also be evaluated in the task of part-of-speech tagging. Honkela, Hyvärinen and Väyrynen (2010) evaluate linguistic features obtained by independent component analysis based on how they can separate sets of words that have different part-of-speech tags.

Especially relevant to our approach are the evaluation methods that apply multilingual vector spaces. Besançon and Rajman (2002) propose an approach in which documents from bilingual corpus are mapped to two separate monolingual vector spaces. The matching documents between the languages are found by comparing the nearest neighbors of each document. The result of the matching is then utilized as an evaluation measure. Further, bilingual vector spaces have been created for lexicon extraction. Gaussier *et al.* (2004) study different methods, including CCA and PLSA, for creating bilingual lexicon from comparable corpora. Sahlgren and Karlgren (2005) evaluate their vector space created from a parallel corpus by comparing the terms in the vector space to bilingual lexica intended for human use. In their evaluation, for each term in the source language w_s, the target-language term w_t given by the system as the closest neighbor is compared to the translations the lexica give for w_s.

3.3.2 Direct evaluation

Direct evaluation methods analyze a vector space by measuring similarities and dissimilarities between feature vectors and comparing them with external data. Usually the idea is to study whether the vector space encodes information on specific semantic relations, such as synonyms, antonyms, sub-, or superconcepts. Thus, many direct evaluation methods can be considered as evaluation in a semantically oriented task that does not require any other components than the vector space itself. As the external data, they often utilize corpora intended for human use, such as lexica, priming data, association norms, or synonym and antonym tests (Sahlgren 2006a).

[1] http://trec.nist.gov/
[2] http://www.clef-campaign.org/
[3] http://research.nii.ac.jp/ntcir/

One common aspect to consider are the paradigmatic and syntagmatic associations (Rapp 2002; Sahlgren 2006b).

The evaluation of vector spaces using the Test for English as a Foreign Language (TOEFL) was first proposed by Landauer and Dumais (1997). The test consists of eighty test items, each having a sentence and four alternative words for one of the words in the sentence. The task is to choose the semantically closest alternative word with respect to the word in the sentence. The best automatic methods perform better than non-native speakers of English on TOEFL test (Rapp 2004). The idea of utilizing language tests has been widely adopted later on. Other tests used for evaluation purposes include, for instance, the Test of English as a Second Language (ESL) multiple-choice synonym questions (Turney 2001), and the SAT (Scholastic Aptitude Test) college entrance exam (Turney 2005). The number of test items in the language tests range from tens to some hundreds.

In addition to the language tests, thesauri have been also used for evaluating vector spaces. The sizes of the thesauri range from thousands to tens of thousand head terms, and thus the coverage is larger than in the language tests. University of South Florida (USF) free association norms of normed words and their associations (Nelson, McEvoy and Schreiber 1998) have been used, for example, by Steyvers, Shiffrin and Nelson (2005). Another association data set is the Edinburgh Associative Thesaurus (EAT) (Kiss *et al.* 1973). Moby synonyms and related terms (used, e.g., by Curran and Moens 2002; Sahlgren 2006b; Väyrynen, Lindqvist and Honkela 2007) is a thesaurus comprising circa 30,000 head terms and a large synonym list for each term. Other thesauri are, for instance, the Macquarie Thesaurus of Australian English (Bernard 1990) and Roget's Thesaurus (Roget 1911). Likewise, more structured lexical databases are available, including the currently widely used WordNet (Fellbaum 1998) and other ontologies of different areas and languages.

Another approach for analyzing the quality of a vector space is to have human evaluators judge the similarity of the vectors close to each other in the vector space (Mitchell and Lapata 2008; Zesch and Gurevych 2009). While human judgement is a very good way of evaluation and can deal with variation within the language users given a large enough number of evaluators, such extensive use of human labor is not feasible to arrange in general. One more approach for direct evaluation is to rely on the studies of meaning representations in humans. For example, Lund and Burgess (1996) show that the semantic distances between words in a vector space correlate with human reaction times in a lexical priming study.

3.3.3 Advantages and drawbacks

In indirect evaluation, a vector space is created for a specific application, which is then used to evaluate the performance of the vector space. The indirect methods do not usually focus on a specific linguistic phenomenon, such as synonymy, but deal with any similarity between the vectors. In general, the performance in one application may not generalize to other applications. Naturally, results of two indirect evaluations are likely to agree when the respective applications benefit from the same aspects of the vector similarity.

On the contrary, direct methods often focus on a specific phenomenon and try to be independent from any specific application. The results may still not generalize to the application evaluations: the fact that a vector space contains a particular type of semantic relation, such as synonymy, does not tell how well it encodes meaning in general (Sahlgren 2006a). Although our proposed method directly evaluates a vector space independent of applications, it does not concentrate on a specific linguistic phenomenon and thus, in this aspect, resembles indirect evaluation methods, without suffering from their limitations.

The main problem of the indirect methods is that they are often time-consuming and need additional components or resources besides the vector space. Direct evaluation methods, including the proposed one, are more straightforward. Some of the indirect methods also resemble direct evaluations in that they are directly based on the similarity of items in the vector space. They include the methods that use bilingual document collections or lexicons (Besançon and Rajman 2002; Gaussier *et al.* 2004; Sahlgren and Karlgren 2005), and are also the methods closest to our proposed approach.

The use of direct evaluation methods suffers from limited evaluation data. Since vector spaces describe semantic similarity, it is natural to use language tests, such as TOEFL, as the reference. However, as the tests are designed for humans, the amount of test data is often small. Thesauri provide larger evaluation sets than the language tests, but since they are also created manually, the availability for a particular domain may be limited. Furthermore, for many languages neither thesauri nor language tests are available. Compared to other direct methods, the proposed evaluation method has no serious problems of data availability: the only required resource is a parallel corpus. Parallel corpora are readily available for several languages and domains, and the amount of suitable data is increasing.

4 CCA-based evaluation of vector representations

In this section, we describe the proposed evaluation method in detail. First, we describe a model of language generation that explains the general idea behind the method and the necessary assumptions. Then we consider the details of a practical evaluation system. Finally, we show two examples with artificial data sets.

4.1 Mathematical foundation

Let $p(s)$ be a probability distribution over documents s in one language. We assume that there exists a D_s-dimensional semantic space, denoted by $\mathcal{Z}_s$, where the meanings of the documents can be encoded, and process $\mathcal{G}_s$ that generates the instances of s from the instances $\mathbf{z}_s \in \mathcal{Z}_s$. Similarly, documents t in another language are generated from D_t-dimensional $\mathbf{z}_t \in \mathcal{Z}_t$ using process $\mathcal{G}_t$. Furthermore, we assume that the semantic spaces for the two languages are subspaces in a global D_z-dimensional semantic space $\mathcal{Z}$, and $\mathbf{z}_s$ and $\mathbf{z}_t$ are linearly dependent on instances of $\mathbf{z}$. That is, given a meaning $\mathbf{z} \in \mathcal{Z}$, documents s and t are produced as

$$s = \mathcal{G}_s(\mathbf{z}_s) = \mathcal{G}_s(\mathbf{W}_s\mathbf{z}) \qquad t = \mathcal{G}_t(\mathbf{z}_t) = \mathcal{G}_t(\mathbf{W}_t\mathbf{z}) \tag{12}$$

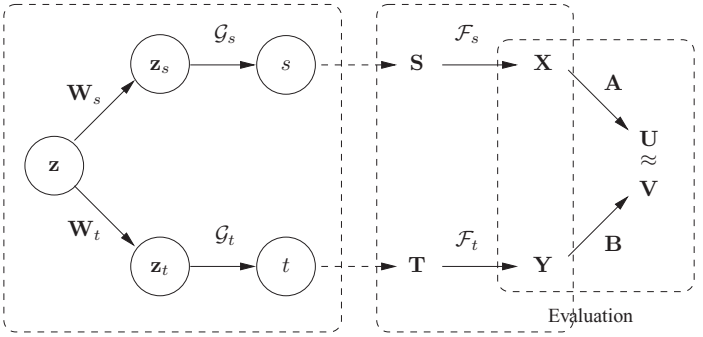

Fig. 1. On the left: Assumed model for generation of documents s and t. Vector $\mathbf{z}$ in the language-independent semantic space $\mathcal{Z}$ is projected onto vectors $\mathbf{z}_s$ and $\mathbf{z}_t$ in the language-specific subspaces $\mathcal{Z}_s$ and $\mathcal{Z}_t$. Processes $\mathcal{G}_s$ and $\mathcal{G}_t$ generate document pairs from the respective subspaces. On the right: The process of evaluating feature extraction method $\mathcal{F}$ with CCA. The aligned document collections $\mathbf{S}$ and $\mathbf{T}$ are reduced to matrices $\mathbf{X}$ and $\mathbf{Y}$ of feature vectors using $\mathcal{F}$. Then $\mathbf{X}$ and $\mathbf{Y}$ are projected onto a common vector space using CCA.

where $\mathbf{W}_s \in \mathbb{R}^{D_s \times D_z}$ and $\mathbf{W}_t \in \mathbb{R}^{D_t \times D_z}$ are rank D_s and rank D_t matrices, respectively. Assuming that $\mathcal{G}_s$ and $\mathcal{G}_t$ are independent processes, also s and t are independent when conditioned on $\mathbf{z}$,

$$p(s, t \mid \mathbf{z}) = p(s \mid \mathbf{z})p(t \mid \mathbf{z}) \tag{13}$$

That is, the only thing that they have in common is their meaning, encoded in $\mathbf{z}$. See the left part of Figure 1 for a graphical illustration of the assumed process of generation.

Let us now consider a feature extraction method $\mathcal{F}$ for the languages. Given two data sets of N documents $\mathbf{S}$ and $\mathbf{T}$ so that each s_i and t_i are samples from $p(s \mid \mathbf{z}_i)$ and $p(t \mid \mathbf{z}_i)$, $\mathcal{F}$ transforms the documents into matrices $\mathbf{X}$ and $\mathbf{Y}$:

$$\mathbf{X} := \mathcal{F}_s(\mathbf{S}) \in \mathbb{R}^{D_x \times N} \qquad \mathbf{Y} := \mathcal{F}_t(\mathbf{T}) \in \mathbb{R}^{D_y \times N} \tag{14}$$

If CCA is applied to $\mathbf{X}$ and $\mathbf{Y}$, it will find projection matrices $\mathbf{A}$ and $\mathbf{B}$ that map $\mathbf{X}$ and $\mathbf{Y}$ to a common vector space as $\mathbf{U} = \mathbf{A}^{\mathrm{T}}\mathbf{X}$ and $\mathbf{V} = \mathbf{B}^{\mathrm{T}}\mathbf{Y}$, respectively, where $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{\min(D_x, D_y) \times N}$. As explained in Section 2, CCA provides orthogonal $\mathbf{U}$ and $\mathbf{V}$ for which the row vectors have the highest correlations $\rho_i = \mathbf{u}_i \mathbf{v}_i^{\mathrm{T}}$.

Using the assumed model of document generation with the original semantic document representations $\mathbf{Z}$, we have

$$\mathbf{U}\mathbf{V}^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}} \mathcal{F}_s(\mathcal{G}_s(\mathbf{W}_s\mathbf{Z}))\mathcal{F}_t(\mathcal{G}_t(\mathbf{W}_t\mathbf{Z}))^{\mathrm{T}}\mathbf{B}_j. \tag{15}$$

Intuitively, any feature from $\mathcal{F}$ that does not originate from $\mathbf{Z}$ will decrease the correlations. In case the projections $\mathbf{W}_s$ and $\mathbf{W}_t$ do not lose any information (i.e., $D_s = D_t = D_z$ and the matrices are invertible), we can show that when the feature extraction method is able to transform the documents back into the semantic spaces $\mathcal{Z}_s$ and $\mathcal{Z}_t$, it provides the highest possible correlations: Inserting $\mathcal{F}_s(\mathcal{G}_s(\mathbf{z}_s)) = \mathbf{z}_s$ and

$\mathcal{F}_t(\mathcal{G}_t(\mathbf{z}_t)) = \mathbf{z}_t$ to (15) results in

$$\mathbf{U}\mathbf{V}^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}}\mathbf{W}_s\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{W}_t^{\mathrm{T}}\mathbf{B} \tag{16}$$

As the matrices $\mathbf{W}_s$ and $\mathbf{W}_t$ are invertible, we can simply set $\mathbf{A}^{\mathrm{T}} = \mathbf{W}_s{}^{-1}$ and $\mathbf{B}^{\mathrm{T}} = \mathbf{W}_t{}^{-1}$, which gives $\mathbf{U}\mathbf{V}^{\mathrm{T}} = \mathbf{Z}\mathbf{Z}^{\mathrm{T}}$. Trivially, this leads to $\rho_i = \mathbf{z}_i\mathbf{z}_i^{\mathrm{T}}/\mathbf{z}_i\mathbf{z}_i^{\mathrm{T}} = 1$ for all i. Even if $\mathbf{W}_s$ and $\mathbf{W}_t$ are not square or full rank matrices, CCA gives the optimal solution for the corresponding eigenvalue problem. Thus, the evaluation, illustrated on the right part of Figure 1, should give insight on how well the tested methods extract features that correspond to the common meaning of the documents.

4.2 Evaluation setup

In the conventional evaluation setup, the learning algorithm – here a feature extraction method – is evaluated. One data set is needed for training the model (training set) and another for evaluating it (evaluation set). In our case, however, the evaluation method includes learning, i.e., calculating CCA. Learning the parameters of CCA either based on the training set of the feature extraction or the final evaluation set would enable over-fitting. Therefore, the evaluation set is divided into two distinct sets: an evaluation training set (evaltrain) and an evaluation test set (evaltest). The evaltrain set is used for training the parameters of CCA and the evaltest set as a test set for estimating the final correlations.[4]

The requirement of the evaluation training set results in a three-stage evaluation setup as illustrated in Figure 2. At the first stage, the feature extraction method is trained for both the languages, using the monolingual training data sets $\mathbf{S}_0$ and $\mathbf{T}_0$. Both the evaltrain data $(\mathbf{S}, \mathbf{T})$ and the evaltest data $(\widetilde{\mathbf{S}}, \widetilde{\mathbf{T}})$ are run through the feature extraction to obtain the aligned sets $(\mathbf{X}, \mathbf{Y})$ and $(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})$.

At the second stage, the evaluation training set is used to calculate CCA as described in Section 2, resulting in the projection matrices $\mathbf{A}$ and $\mathbf{B}$, the projected data sets $\mathbf{U}$ and $\mathbf{V}$, and the correlations $\boldsymbol{\rho} = [\rho_1, \ldots, \rho_D]$. As a regularization, we add a small positive value ϵ proportional to the variances of $\mathbf{X}$ and $\mathbf{Y}$ to the diagonals of the respective covariance matrices $\mathbf{C}_{xx}$ and $\mathbf{C}_{yy}$:

$$\widehat{\mathbf{C}}_{xx} = \mathbf{C}_{xx} + \epsilon\,\mathbf{S}_x \qquad \widehat{\mathbf{C}}_{yy} = \mathbf{C}_{yy} + \epsilon\,\mathbf{S}_y \tag{17}$$

where $\mathbf{S}_x$ is a diagonal matrix with $S_{x(ii)} = {\sigma_{x_i}}^2$, $\mathbf{S}_y$ is a diagonal matrix with $S_{y(ii)} = {\sigma_{y_i}}^2$, and $0 < \epsilon \ll 1$.

At the third stage we estimate how the learned features and the CCA projections together generalize to new data. Especially if the number of samples in the evaltrain set is low, or the dimensionalities of $\mathbf{X}$ and $\mathbf{Y}$ are high, the sample estimates of the covariance matrices are not robust. This leads to overlearning of the projection matrices $\mathbf{A}$ and $\mathbf{B}$, regardless of the regularization. To find out how the learned projections can generalize outside the evaltrain set, we use them to project the evaltest set into the same common space, and calculate correlations for the resulting

[4] Another view, pointed out by one of the reviewers, is that our setup also evaluates the evaluation method (CCA).

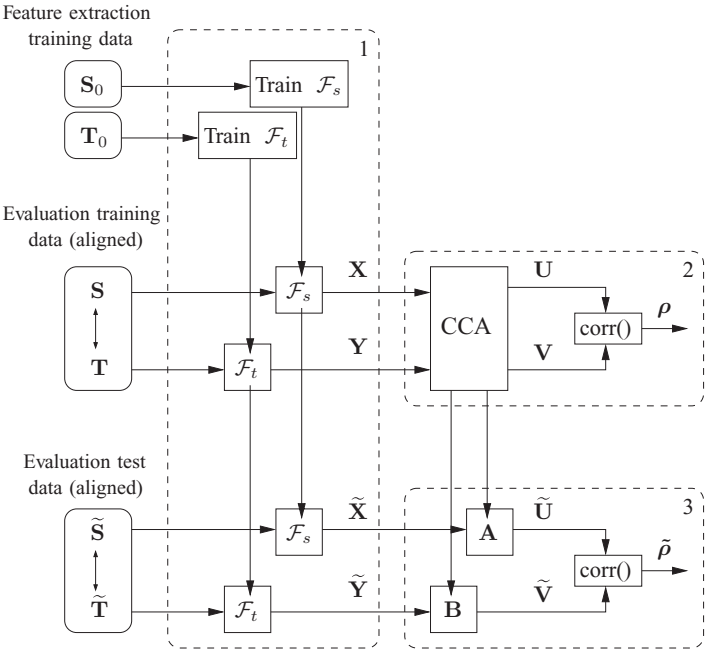

Fig. 2. Diagram of the evaluation setup. (1) The feature extraction method $\mathcal{F}$ is trained on monolingual corpora and then applied to transform the evaluation data sets into vectorial form. (2) CCA is trained on the evaluation training data to find the canonical variates $\mathbf{U}$ and $\mathbf{V}$ and the respective projection matrices $\mathbf{A}$ and $\mathbf{B}$. (3) The evaluation test data are then projected into the same space, and finally the test set correlations $\tilde{\boldsymbol{\rho}}$ are computed.

matrices. Assuming that $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$ are centered, the test set correlations $\tilde{\rho}_i$ are calculated as follows:

$$\tilde{\rho}_i = \frac{\mathbf{a}_i^{\mathrm{T}} \widetilde{\mathbf{X}} \widetilde{\mathbf{Y}}^{\mathrm{T}} \mathbf{b}_i}{\sqrt{\mathbf{a}_i^{\mathrm{T}} \widetilde{\mathbf{X}} \widetilde{\mathbf{X}}^{\mathrm{T}} \mathbf{a}_i} \sqrt{\mathbf{b}_i^{\mathrm{T}} \widetilde{\mathbf{Y}} \widetilde{\mathbf{Y}}^{\mathrm{T}} \mathbf{b}_j}} \tag{18}$$

The vector of the test set correlations, $\tilde{\boldsymbol{\rho}}$, is used to obtain final score for the feature extraction method. The evaluation measures are discussed in the next subsection.

The covariance matrices $\mathbf{C}_{xx}$, $\mathbf{C}_{yy}$, and $\mathbf{C}_{xy}$, which are needed in computing the canonical correlations, are non-sparse and thus the memory usage is of magnitude $O({D_x}^2 + {D_y}^2 + D_x D_y)$. In consequence, we need to keep the dimensionalities D_x and D_y low. It is convenient, but not necessary, to use representations that have the same dimensionality D for both languages. By performing the evaluation for a range of values for D, one may need to find the optimal dimensionality for the evaluated feature extraction method given the evaluation measure.

4.3 Evaluation measures

Learning the optimal projections with CCA and using them on the evaltest set gives us the correlation estimates $\tilde{\boldsymbol{\rho}}$. To make the evaluation more straightforward, we prefer to have a single value that measures the quality of a representation.

In the simplest case, we want to compare two vector representations having the same number of features returned by two feature extraction methods. Since the task is not to find a subset of the features, we do not consider only the largest correlation or the sum of few largest correlations. Instead, an intuitive measure is the sum over all the correlations:

$$R(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}}) = \sum_{i=1}^{D} \tilde{\rho}_i \tag{19}$$

For perfectly correlated sets, $R = D$, and for uncorrelated sets, $R = 0$.

Canonical correlation analysis restricts the learned evaltrain correlations to be positive. However, the correlations $\tilde{\rho}_i$ from the evaltest set can also be negative due to random variation between the evaltrain and evaltest sets. It is justifiable that negative correlation coefficients decrease the score: such a coefficient for the test set indicate that CCA has learned something that does not generalize outside the evaltrain set. Moreover, forcing $\tilde{\rho}_i$ to be positive, for example, by taking the absolute value, would introduce a bias to its expected value.

Theoretically, MI would be a natural choice for an evaluation measure. However, as a general measure of dependence, MI can be inferred from the correlations only when the data are normally distributed. Still, even when we know that the data do not follow a Gaussian distribution, we can use (8) for MI to obtain a 'Gaussian–MI' score $G(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})$. For uncorrelated sets, $G(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}}) = 0$. An evident difference to the sum of correlations is that the Gaussian–MI score will give more weight to the correlations that have absolute values close to one. In fact, already one $\tilde{\rho}_i$ that is exactly one will set the score to infinity. Because of the squared correlation coefficients, negative values increase the score. However, as high negative values are very improbable, the effect is small in practice.

If the evaluated feature extraction methods return different number of features, the comparison of the vector representations is not straightforward. A problem with both the correlation sum and the Gaussian–MI score is that the scores tend to increase with the number of dimensions. An option would be to consider, for instance, the average correlation $R(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})/D$. While the average correlation would directly penalize for having uncorrelated features, we find it unintuitive that, for example, the result $\tilde{\boldsymbol{\rho}} = [0.9, 0.8]$ ($D = 2$) would be worse than $\tilde{\boldsymbol{\rho}} = [0.9]$ ($D = 1$), as the representation in the former case surely encodes more semantic information than in the latter. Moreover, even a dimension that has a very small positive correlation can be useful if it is weighted according to the strength of the correlation, as shown by Tripathi *et al.* (2008). We compare representations with different number of dimensions, first using an artificial example in Section 4.4 and later when validating the evaluation results with two sentence matching tasks in Section 5.3.

4.4 Examples

We demonstrate the proposed evaluation method on two artificial data sets. The goal is to show the effects of noise and dimensionality on the two evaluation measures, i.e., the sum of correlations and the Gaussian–MI score. In addition, the examples justify the need for separate test data set in the evaluation setup.

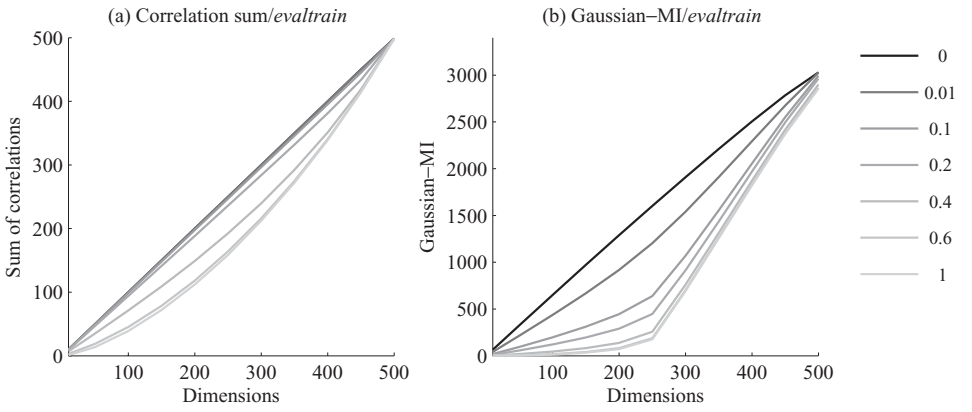

Fig. 3. Effect of different noise levels for evaluation training data: (a) Correlation sum $R(\mathbf{X}, \mathbf{Y})$, and (b) Gaussian–MI score $G(\mathbf{X}, \mathbf{Y})$. Lighter the tone of the curve, higher the noise level.

Example 1: *Examining the effect of noise*. Here we show how noise affects the evaluation scores. Consider the following normally distributed data with additive noise:

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{20}$$

$$\mathbf{n}_x \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{21}$$

$$\mathbf{n}_y \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{22}$$

$$\mathbf{x} := (1 - \alpha)\mathbf{z} + \alpha\mathbf{n}_x \tag{23}$$

$$\mathbf{y} := (1 - \alpha)\mathbf{z} + \alpha\mathbf{n}_y \tag{24}$$

If the proportion of noise $\alpha = 1$, then $\mathbf{x}$ and $\mathbf{y}$ are independent and the expected values for the correlation sum and Gaussian–MI scores are zero. If there is no noise at all ($\alpha = 0$), then $\mathbf{x}$ and $\mathbf{y}$ have a perfect linear dependence. Using our regularization with coefficient ϵ, this results in expected correlation coefficients $1/(1 + \epsilon)$. Thus,

$$R(\mathbf{x}, \mathbf{y}) = \frac{D}{1 + \epsilon} \approx D \times (1 - \epsilon) \tag{25}$$

$$G(\mathbf{x}, \mathbf{y}) = -\frac{D}{2} \times \ln \frac{\epsilon^2 + 2\epsilon}{(\epsilon + 1)^2} \approx -\frac{D}{2} \times \ln(2\epsilon) \tag{26}$$

R approaches D and G approaches infinity as ϵ approaches zero.

We used 500 samples of $\mathbf{z}$ as development data and 500 samples of $\mathbf{z}$ as test data. The test data was divided into fifty subsets. As we did not do feature extraction, training data were not needed. The dimensionality D was varied from 0 to 500 and the level of noise α from 0 to 1. In the evaluation, we applied regularization with $\epsilon = 10^{-6}$. We computed the sum of correlations and the Gaussian–MI scores for evaluation training and test data. In Figures 3 and 4 we use median (second quartile) and first and third quartiles to indicate the central tendency and variability of the results on the evaltest data.

The correlations in the evaltrain data are high even with very noisy data (Figure 3). Especially if D is increased and the number of samples is fixed, the sample

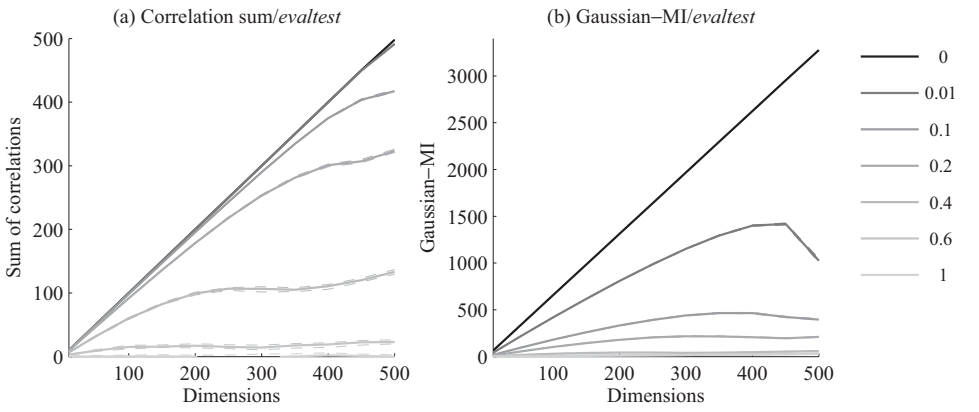

Fig. 4. Effect of different noise levels for evaluation test data: (a) Correlation sum $R(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})$, and (b) Gaussian–MI score $G(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})$. Medians are drawn with solid lines. For consistency, also the first and third quartiles (dashed lines) are drawn, although these are very close to the median in these figures.

covariances become poor estimates for real covariances, and CCA overlearns. The use of the evaltest data for estimating the correlation clearly prevents the problem (Figure 4). As high correlations are obtained only when the learned projections generalize to new data, increasing D cannot improve the scores of the noisy data sets.

Example 2: *Discovering the intrinsic dimensionality*. Next, we consider data in which the intrinsic dimensionality is lower than those of the observed variables. The idea is to show that the evaluation method is able to detect the correct dimensionality given a suitable dimensionality reduction method. Assume that M-dimensional samples, $\mathbf{x}$ and $\mathbf{y}$, are produced from a latent K-dimensional variable $\mathbf{z}$, where $K < M$ is the intrinsic dimensionality, as follows:

$$\mathbf{x} = (1 - \alpha)\mathbf{W}_x\mathbf{z} + \alpha\mathbf{n}_x \tag{27}$$

$$\mathbf{y} = (1 - \alpha)\mathbf{W}_y\mathbf{z} + \alpha\mathbf{n}_y \tag{28}$$

Again, we use normal distributions with zero mean and unit variance for $\mathbf{z}$, $\mathbf{n}_x$, and $\mathbf{n}_y$. Let the weights in $\mathbf{W}_x \in \mathbb{R}^{M\times K}$ and $\mathbf{W}_y \in \mathbb{R}^{M\times K}$ be uniformly distributed between $[-0.5$ and $0.5]$. Without noise, the maximal correlations for the D-dimensional features are $R(\mathbf{x}, \mathbf{y}) \approx \min(D, K) \times (1 - \epsilon)$ and $G(\mathbf{x}, \mathbf{y}) \approx -\min(D, K)/2 \times \ln(2\epsilon)$ and thus remain constant for $D > K$.

We compare PCA, which is a standard feature extraction method for the Gaussian data, and a trivial feature selection method that selects an arbitrary subset of features in $\mathbf{x}$ and $\mathbf{y}$. We used 500 samples of $\mathbf{z}$ as a training set (i.e., for calculating the projections of PCA), another 500 samples as an evaluation training set, and once more 500 samples as an evaluation test set. Other parameters were $\alpha = 0.5$, $K = 50$, and $M = 250$. Figure 5 shows the scores for the evaltrain data and Figure 6 for the evaltest data. The results show that PCA (gray line) performs considerably better than the baseline by just selecting a subset of variables (black line) when the dimensionality is near to the number of latent dimensions (50). Due to overlearning,

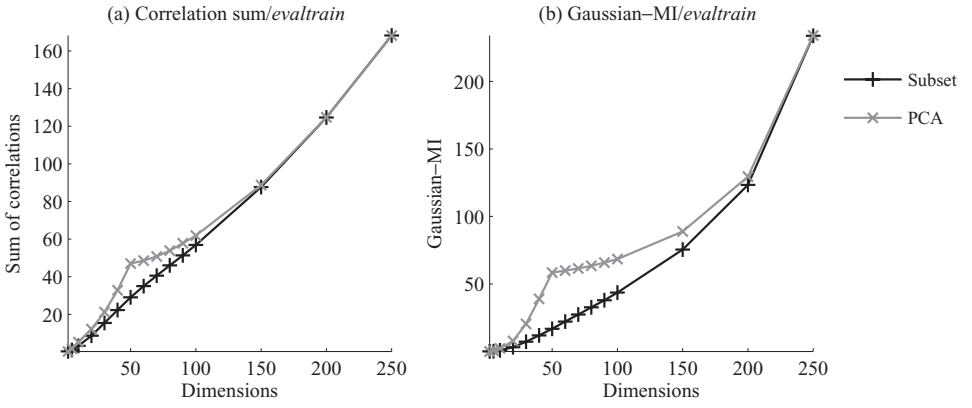

Fig. 5. Evaluation training set results for data with 50 intrinsic dimensions: (a) Correlation sum, and (b) Gaussian–MI score. In subset, a random subset of D dimensions of $\mathbf{X}$ and $\mathbf{Y}$ are selected. In PCA, the D first principal components are applied.

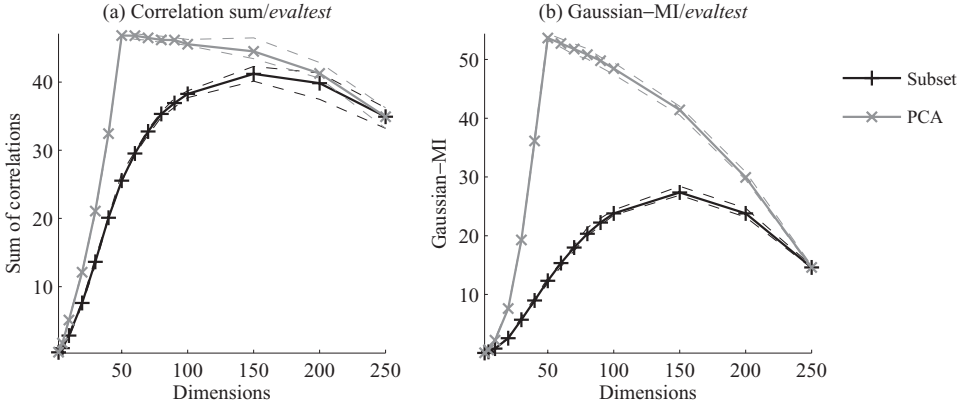

Fig. 6. Evaluation test set results for data with 50 intrinsic dimensions: (a) Correlation sum, and (b) Gaussian–MI score. In subset, a random subset of D dimensions of $\mathbf{X}$ and $\mathbf{Y}$ are selected. In PCA, the D first principal components are applied. Medians are drawn with solid lines and the first and third quartiles with dashed lines.

the correlations for the evaltrain data continue to grow with dimensionality. For the evaltest data, however, they slowly start to decrease, thus clearly indicating the correct number of intrinsic dimensions.

5 Experiments

In the experiments, we validate the proposed evaluation method in three different ways. First, we confirm that the results based on our evaluation are in agreement with previous findings regarding vector space models for language. Second, we compare the results of the evaluation method with the results of indirect evaluations in two sentence-matching tasks. Third, we compare the results against a quantitative manual evaluation of factor loadings of bilingual features found by CCA.

Table 2. *The number of sentences, word occurrences, and word forms in the English–Finnish data sets*

Name	Dates	Sentences	Word occurrences		Word forms	
			English	Finnish	English	Finnish
Training sets						
Day	2003-09-25	1,356	30,550	23,562	4,058	8,116
Month	2003-09	21,358	496,044	382,866	16,172	51,550
Year	2003	100,801	2,344,282	1,816,663	31,124	129,273
Evaluation sets						
Evaltrain	2000-09	12,172	304,090	218,903	12,103	34,937
Evaltest	2000-Q4	1,755	17,968	13,991	3,210	4,833

5.1 Data sets

We used the Europarl corpus (version 2) that consists of the proceedings of the European Parliament meetings in eleven European languages (Koehn 2005). We applied standard preprocessing for the data that lowercased all letters and separated punctuation marks from words. The major parts of the experiments were performed with English–Finnish data, but Danish, German, and Swedish were also used. We extracted data sets of three different sizes for training feature extractions: one day of the corpus (2003-09-25), one month (2003-09), and one year (2003). As an evaluation training set, we applied the sentences that contained five to fifteen words in both the languages from the month 2000-09. As an evaluation test set, we used 1,755 sentences extracted by Koehn, Och and Marcu (2003) from the last quarter of year 2000. The sentences contained five to fifteen words that were same for each language in the corpus. We divided the test set into ten parts and used the one-sided right-tailed Wilcoxon signed-rank test with significance level 0.05 to calculate statistical significances. The number of sentences, word occurrences, and different word forms are shown in Table 2. For practical reasons, we used the one-month training set for most of the experiments: It is small enough for fast calculation of different dimensionality reductions, but large enough to keep the out-of-vocabulary rates for the data sets used in the evaluation within reasonable limits.

To find out whether the selected evaltrain data set was large enough for learning good correlation estimates, we run one feature extraction, SVD for plain cooccurrence counts, with different target dimensionalities for different sizes of the set. Figure 7 shows the estimated scores both from evaltrain and evaltest sets. The higher the dimensionality, the more data required before the estimates converge. Moreover, the gap between the evaltrain and evaltest results increases, showing that the high-dimensional features are less robust for the variation of the data. The full 12,000-sentence data can be considered large enough for reliable CCA results up to 1,000-dimensional representations.

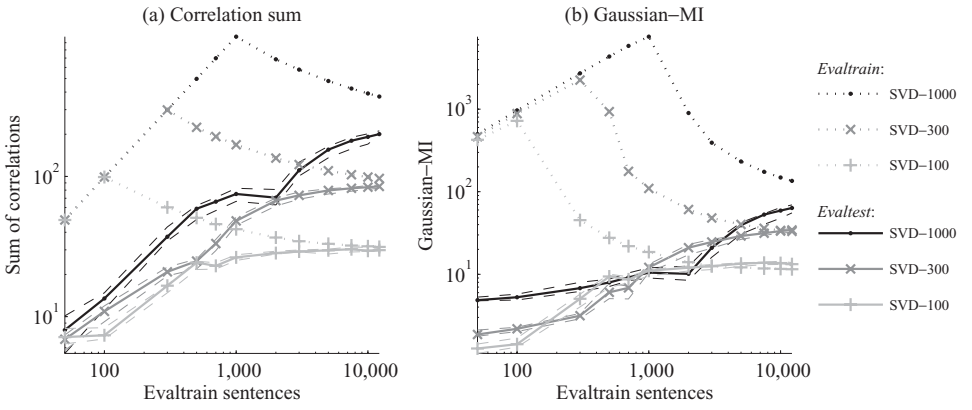

Fig. 7. Evaluation test and training set results for 100-, 300-, and 1,000-dimensional SVD features for different sizes of evaltrain set: (a) Correlation sum, and (b) Gaussian–MI score. Solid lines show the evaltest scores and dotted lines the evaltrain scores. The peaks in the evaltrain scores occur when the number of samples equals the number of dimensions.

5.2 Validation with known facts

A plausible evaluation method should be able to re-establish previously known facts concerning vector space models. We therefore consider five hypotheses from the literature and examine whether our evaluation method confirms them.

(1) *Dimensionality reduction*: SVD and PCA should perform better than other dimensionality reduction methods when the target dimensionality is relatively low (Deerwester *et al.* 1990; Kaski 1998; Bingham and Mannila 2001).
(2) *Weighting*: Among different global weighting schemes, entropy weighting and logarithmic idf should give the best improvements when using SVD (Dumais 1991; Nakov, Popova and Mateev 2001).
(3) *Amount of data*: More training data should improve results whenever a reasonable dimensionality reduction method is applied. (Zelikovitz and Hirsh 2001; Yarowsky and Florian 2002).
(4) *Phrases as features*: Generally, it does not seem to help if observed phrases (e.g., 2-grams) from the training data are included as features (Lewis 1992; Scott and Matwin 1999; Caropreso, Matwin and Sebastiani 2001; Sebastiani 2002; Koster and Seutter 2003; Coenen *et al.* 2007).
(5) *Effect of language similarity*: Due to morphological and syntactic similarities, closely related languages should have higher correlations than, for example, languages that belong to different language families (Besançon and Rajman 2002; Chew and Abdelali 2007; Sadeniemi *et al.* 2008).

Except for the fourth experiment, in which bag-of-phrases are tested, we used the basic bag-of-words representations for the sentences. All the word forms, including punctuation marks, were collected to form the word–sentence matrix $\widehat{\mathbf{X}}$, where $\hat{\mathbf{x}}_{ij}$ is the number of times word i occurred in sentence j. Except for the first experiment, the matrix was weighted according to global and local weightings, and the dimensionality

Table 3. *The evaluated dimensionality reduction methods.* $\widehat{\mathbf{X}}$ *is the original data matrix and* $\mathbf{X}$ *is the reduced data matrix*

Method	Explanation	
W-RandSet	Randomly selected D words	$\mathbf{X} = \widehat{\mathbf{X}}_{R_D}$
W-FreqSet	The most frequent D words	$\mathbf{X} = \widehat{\mathbf{X}}_{F_D}$
W-RandProj	Projecting words to D-dimensional nearly orthogonal vectors	$\mathbf{X} = \mathbf{R}_D\widehat{\mathbf{X}}$
S-RandSet	Inner product with randomly selected D sentences	$\mathbf{X} = \left(\widehat{\mathbf{X}}_{R_D}\right)^{\mathrm{T}}\widehat{\mathbf{X}}$
S-LenSet	Inner product with the longest D sentences	$\mathbf{X} = \left(\widehat{\mathbf{X}}_{L_D}\right)^{\mathrm{T}}\widehat{\mathbf{X}}$
S-RandProj	Inner product with all the sentences projected to D-dimensional nearly orthogonal vectors	$\mathbf{X} = \left(\mathbf{R}_D\widehat{\mathbf{X}}\right)^{\mathrm{T}}\widehat{\mathbf{X}}$
SVD	The first D right singular vectors of the singular value decomposition $\widehat{\mathbf{X}} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$	$\mathbf{X} = \mathbf{V}_D{}^{\mathrm{T}}$

was reduced with SVD. We experimented with a range of dimensionalities D from 10 to 1300.

5.2.1 Dimensionality reduction

We compared seven dimensionality reduction methods that we summarize in Table 3. Three of them were based on the first-order information (see Section 3). We used two feature selection methods: taking a random subset of words (W-RandSet) and taking the words with the highest sentence frequency (W-FreqSet). We also tested random projection for the words (W-RandProj).

As second-order dimensionality reduction, we used the following four methods. The first one is a naive approach, which selects a random subset of sentences (S-RandSet). The next approach is to select sentences that contain the highest number of word types, because intuitively they provide more information (S-LenSet). We also used random projection, where the random vectors are now sampled for each sentence rather than each word (S-RandProj). Finally, we tested SVD.

The left side of Figure 8 shows the correlation sum results. Its right side shows same results using relative differences to SVD and a logarithmic scale for dimensionality, making differences more clear. W-FreqSet and SVD clearly outperformed other methods. SVD was significantly better than W-FreqSet for all dimensionalities except for $D = 10$ (significantly worse), $D = 900$ (significantly worse), and $D = 1,300$ (no statistically significant difference). S-RandProj was usually the best of the other second-order methods, but the difference to S-LenSet was statistically significant only for $D \in \{10, 20, 200, 300\}$.

Figure 9 shows the results for the Gaussian–MI score. The results are similar to the sum of correlations except that W-FreqSet is closer to SVD and even outperforms it at $D = 100$. To illustrate the reason for the conflicting results, Figure 10 gives a closer look at the distribution of test correlations for W-FreqSet and SVD. While SVD has on average higher correlations (as indicated by the higher sum), the first few values ranked from 1 to 13 are higher for W-FreqSet. Since the function $-\ln(1-\rho_i{}^2)$

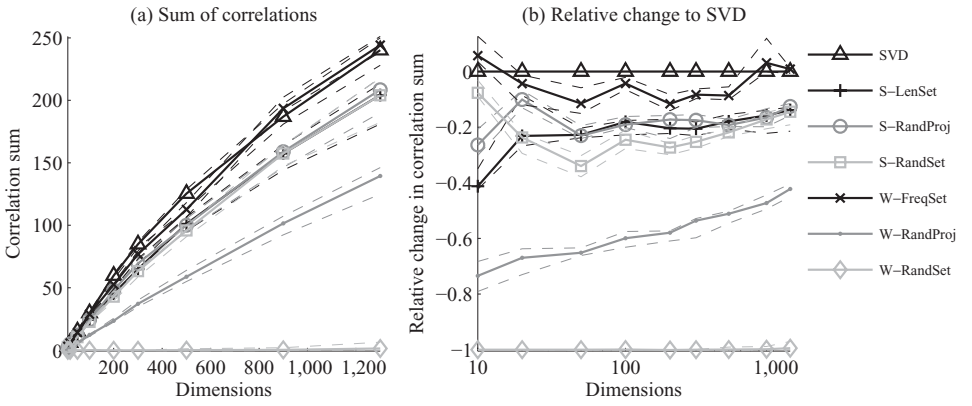

Fig. 8. Results for different dimension reduction methods: (a) Absolute sum of correlations. The dashed lines show the first and third quartiles for the results of each method. (b) Relative change compared to SVD. The reference (SVD) does not have quartile lines, because the relative differences were calculated separately for each subset and the deviation is thus included in the results of the compared methods.

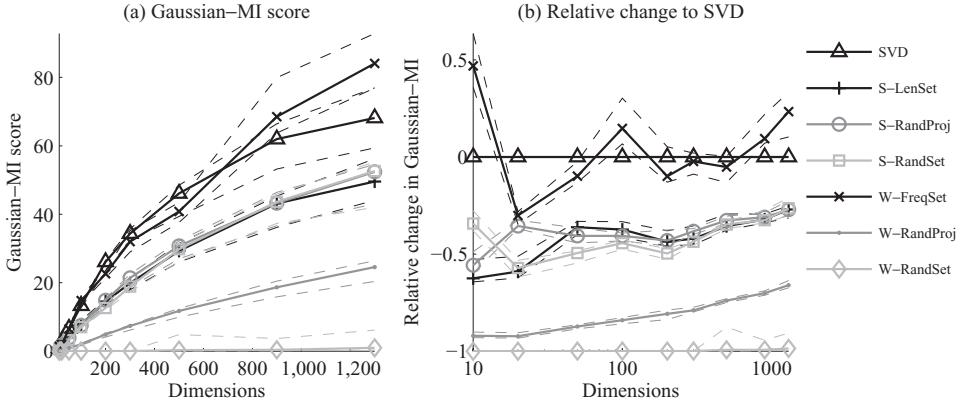

Fig. 9. Results for the different dimension reduction methods: (a) Absolute Gaussian–MI scores, and (b) relative change compared to SVD. The dashed lines show the first and third quartiles for the results.

gives much emphasis to values close to one and little to those near to zero, this explains why W-FreqSet has better Gaussian–MI score than SVD. In the extreme case if there was a single correlation very close to one and the rest were zeros, the Gaussian–MI score would still be high. We used only correlation sum for the remaining experiments to avoid this problem.

To sum up, our evaluation prefers SVD in lower dimensionalities. This is in agreement with previous studies (Deerwester *et al.* 1990). Moreover, the order of other dimensionality reduction methods was intuitive.

5.2.2 Weighting and normalization

In the previous experiment, no feature weighting was applied. In our second experiment, we use SVD for dimensionality reduction and compare the global

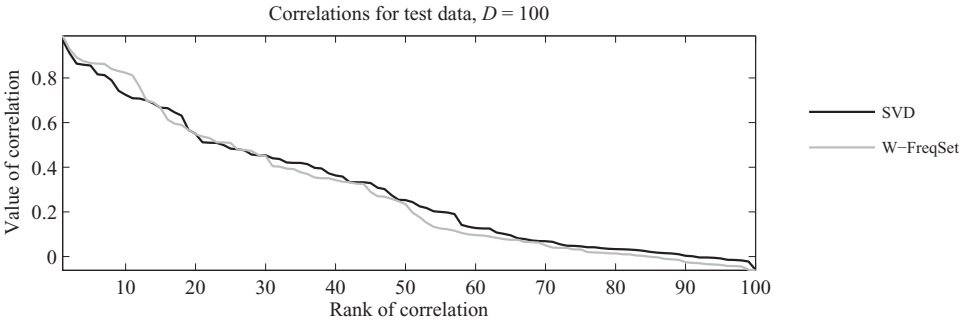

Fig. 10. Individual test correlations for SVD and W-FreqSet dimensionality reductions for $D = 100$. Values of the correlations are shown in descending order.

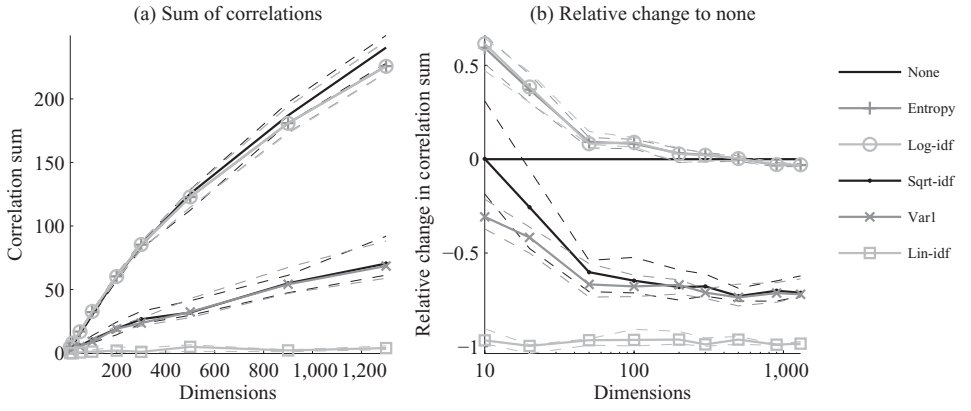

Fig. 11. Results for different global weightings: (a) Absolute sum of correlations, and (b) relative change compared to no weighting. The dashed lines show the first and third quartiles for the results of each method.

term weightings described in Table 1. Figure 11 shows (a) the sum of correlations for the weighting methods, and (b) the relative difference to the baseline without weighting.

Again, the results are in line with the previous findings (Dumais 1991; Nakov *et al.* 2001). The logarithmic idf and entropy weightings improved the baseline results significantly for the lower dimensionalities ($D < 200$). Up to $D = 500$, there was no significant difference, and for $D \geq 1,000$, the weighting decreases the result. We did not find significant differences between logarithmic idf and entropy weightings as found by Dumais (1991). The rest of the weightings gave scores significantly below the baseline. Linear idf was always the worst.

We also tested different local weightings (no, binary, and logarithmic weight) for the term frequencies and length normalizations (no, L1, and L2 normalization) with entropy weighting and SVD. The logarithmic term frequency outperformed plain term frequency significantly for some dimensionalities ($D \in \{20, 50, 100, 1300\}$) and was never worse, as shown in Figure 12. Using binary values was better than logarithmic term frequency (and term frequency) only for $D = 10$. With log-term

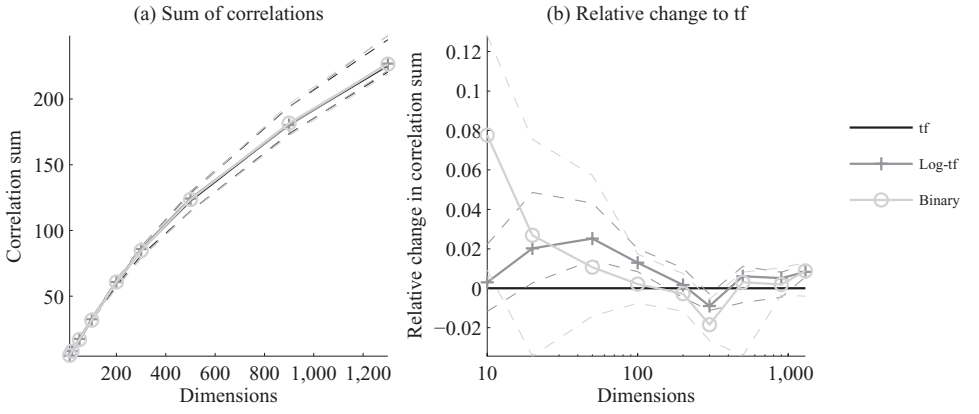

Fig. 12. Results for different local weightings: (a) Absolute sum of correlations, and (b) relative change compared to no weighting. The dashed lines show the first and third quartiles for the results of each method.

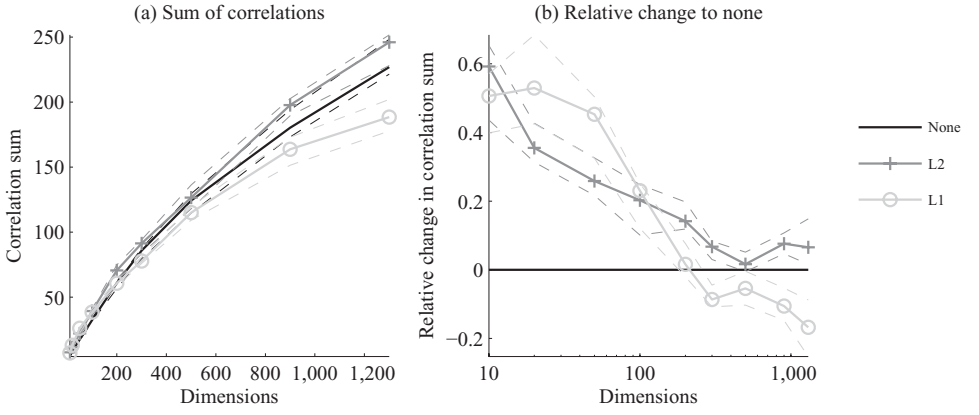

Fig. 13. Results for different length normalizations: (a) Absolute sum of correlations, and (b) relative change compared to no normalization. The dashed lines show the first and third quartiles for the results of each method.

frequency and entropy weighting, L2 normalization was significantly better than no length normalization except for $D = 500$, where the difference was not significant (Figure 13). L1 was significantly better than L2 for some low dimensionalities ($D \in \{20, 50\}$). However, it was even worse than no normalization for high dimensionalities ($D \geq 300$). Thus, the most robust setting for SVD seems to be the entropy weighting as the global weighting, logarithmic term frequency as the local weighting, and L2 normalization of the lengths. We applied this for the rest of the experiments.

5.2.3 Amount of data

As a third experiment, we studied the effect of the size of the training set for feature extraction methods. Figure 14 shows the absolute correlation sums and the relative difference to the results with the one-month data set. The one-year

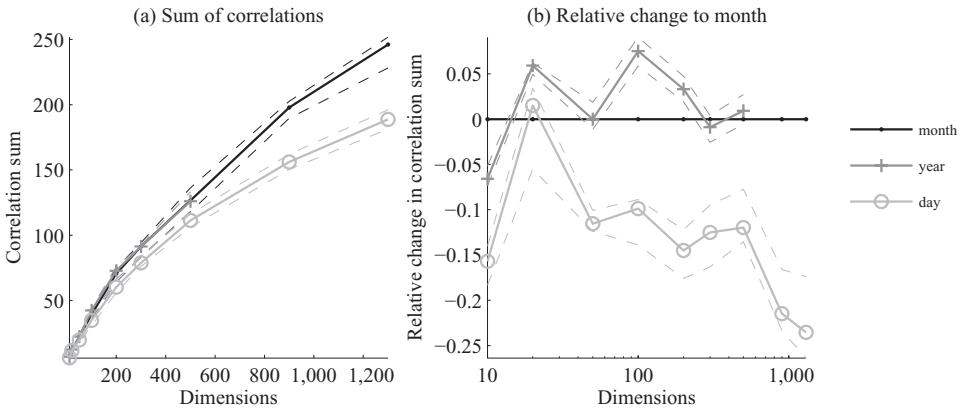

Fig. 14. Results for different sizes of training data for SVD: (a) Absolute sum of correlations, and (b) relative change compared to the one-month data set. For the year data set, we used dimensionality up to 500 only due to high computational requirements.

data set gave significantly better correlation sums than the one-month data set for $D \in \{20, 100, 200\}$ and significantly worse results only for $D = 10$. The one-day data set gave significantly worse results than the larger sets except for $D = 20$, where the difference to the one-month set was not significant. Thus, our evaluation method shows that more training data improve results, which conforms with the findings of, for instance, Zelikovitz and Hirsh (2001) and Yarowsky and Florian (2002). The finding is especially pronounced in higher dimensionalities, which has been already noticed, for example, by Bradford (2008).

5.2.4 *Phrases as features*

Intuitively, the bag-of-words representation may not always be the optimal representation for sentences, because it discards the information on the word order. A simple extension is to use sequences of multiple words, n-grams, as additional features. However, previous work on the bag-of-phrases models shows that including n-grams does not improve the results in tasks, such as text categorization, unless the n-grams are selected carefully (Lewis 1992; Scott and Matwin 1999; Caropreso *et al.* 2001; Sebastiani 2002; Koster and Seutter 2003; Coenen *et al.* 2007).

We tested the evaluation for representations in which the n-grams up to a maximum length n that had been observed in the training data at least twice were included as separate features in addition to all the observed word forms. Table 4 shows the number of features in the English–Finnish data for $n = 2$ and $n = 5$. Similar to the bag-of-words baseline, we applied SVD, log-term frequency and entropy weightings, and L2 normalization to get final representations.

Figure 15 shows (a) the correlation sums for bag-of-words and bag-of-phrases with $n = 2$ and $n = 5$, and (b) the relative changes to bag-of-words with logarithmic scale for dimensionality. Bag-of-words outperformed bag-of-phrases only for very low dimensionalities ($D = 10$ for 2-grams and $D \in \{10, 20\}$ for 5-grams). For $D \geq 1,000$, there was no statistically significant differences. Otherwise, both 2-grams and 5-grams

Table 4. *The amount of n-gram types in the English and Finnish training data. For $n > 1$, n-grams occurring only once are discarded*

Max. length n	English n-grams	Finnish n-grams
1	16,172	51,550
2	68,155	89,046
5	180,068	125,563

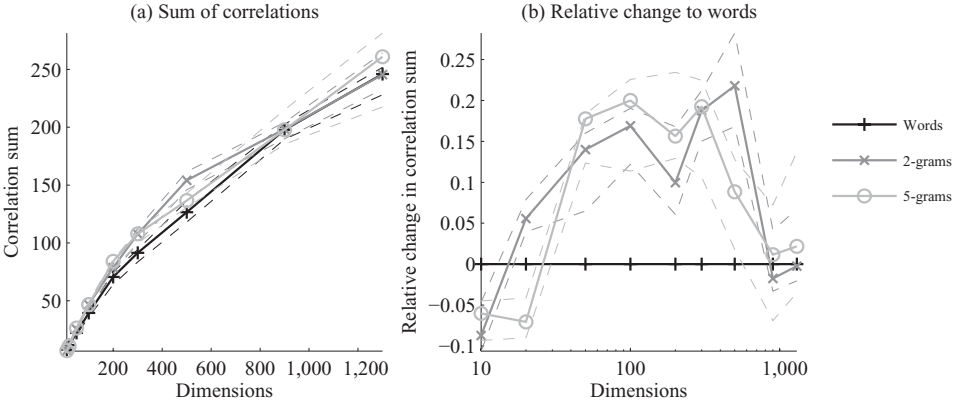

Fig. 15. Correlation sums for test data using n-gram features of different lengths: (a) Absolute sum of correlations, and (b) relative change compared to words.

provided statistically significant improvements for correlation sum. The order of the two was not clear: sometimes 2-grams were better ($D \in \{20, 500\}$), sometimes 5-grams ($D \in \{50, 200\}$), and sometimes there was no significant difference.

Our method for selecting the phrases was very simple. Thus, the results are in contrast with the previous studies, in which only sophisticated selection methods, if any, have provided improvements. One reason may be that the word order information is more important for the meaning of single sentences than for longer documents. This would require further study with a different type of data set.

5.2.5 *Effect of language similarity*

A reasonable hypothesis is that closely related languages will be more correlated than distant languages (Besançon and Rajman 2002; Chew and Abdelali 2007; Sadeniemi *et al.* 2008). The results for the test data using similar bag-of-words (entropy weighting, L2 normalization, and dimensionality reduction with SVD) representations for six language pairs are shown in Figure 16. For the highest dimensionalities ($D \geq 500$), the order of the scores follow the closeness of the languages: (1) Among the language pairs, the most correlated were Danish and Swedish, two closely related North Germanic languages. (2) The languages in the three least correlated pairs belonged to different language families: one of the languages was always Finnish, which is an Uralic language, and the other was English, German,

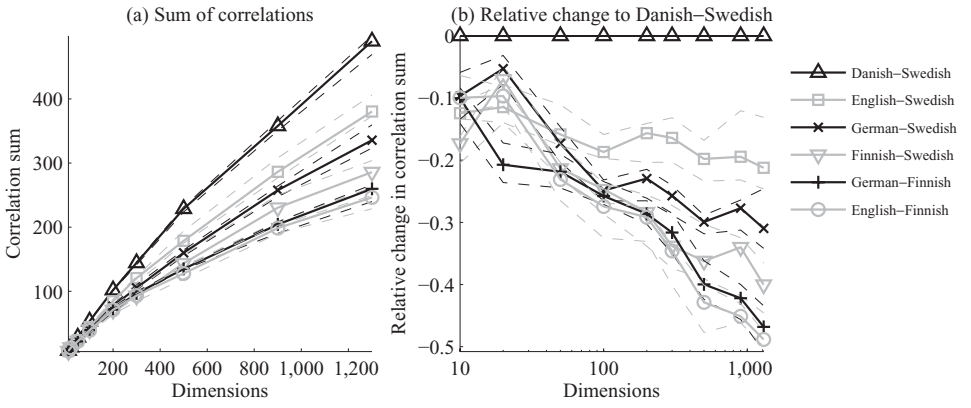

Fig. 16. Correlation sums for test data with different language pairs: (a) Absolute sum of correlations, and (b) relative change compared to Danish–Swedish.

or Swedish, which are all Indo-European and Germanic languages. (3) Finnish was more correlated with Swedish than any other of the Germanic languages, which may be due to the close cultural proximity of speakers of the two languages.

5.3 Validation with sentence matching tasks

An indirect way to evaluate the quality of vector representations for bilingual corpus is to check how well the sentences that are translations of each other can be matched based on a vector space model. We compared our proposed direct approach to an indirect approach of using sentence matching for evaluation. The idea is to show that any vector representation that performs well on our evaluation approach also performs well on the sentence matching task.

We considered the task of matching in two different settings. In the first setting, we used the projected sets **U** and **V** to find the closest sentence in English for a sentence in Finnish and checked if it was a correct translation. We also searched for a match in the other direction and took the average of two accuracies. This task is similar to the nearest translation test done by Besançon and Rajman (2002). In the second setting, we found one-to-one alignment of sentences using the Hungarian algorithm (Kuhn 1955), weighting the dimensions of canonical variates with corresponding evaltrain correlations in a similar manner as done by Tripathi *et al.* (2010).

We first validated the evaluation scores for 100-dimensional representations obtained from the previous experiments: the different dimensionality reductions without weighting, weighted SVD for the one-day, the one-month, and the one-year data sets, and the two bag-of-phrases representations. The test data were divided into 100 subsets of seventeen to eighteen sentences, and both the correlations and matching accuracies were calculated for each set.

Figure 17 shows scatter plots of accuracy in the nearest translation test (top) and the alignment task (bottom) versus the correlation sum. For the translation task, Spearman's rank correlation coefficient between the correlation sums and accuracies was 0.776 over the subsets and 1.0 over the means. For the alignment task, correlation

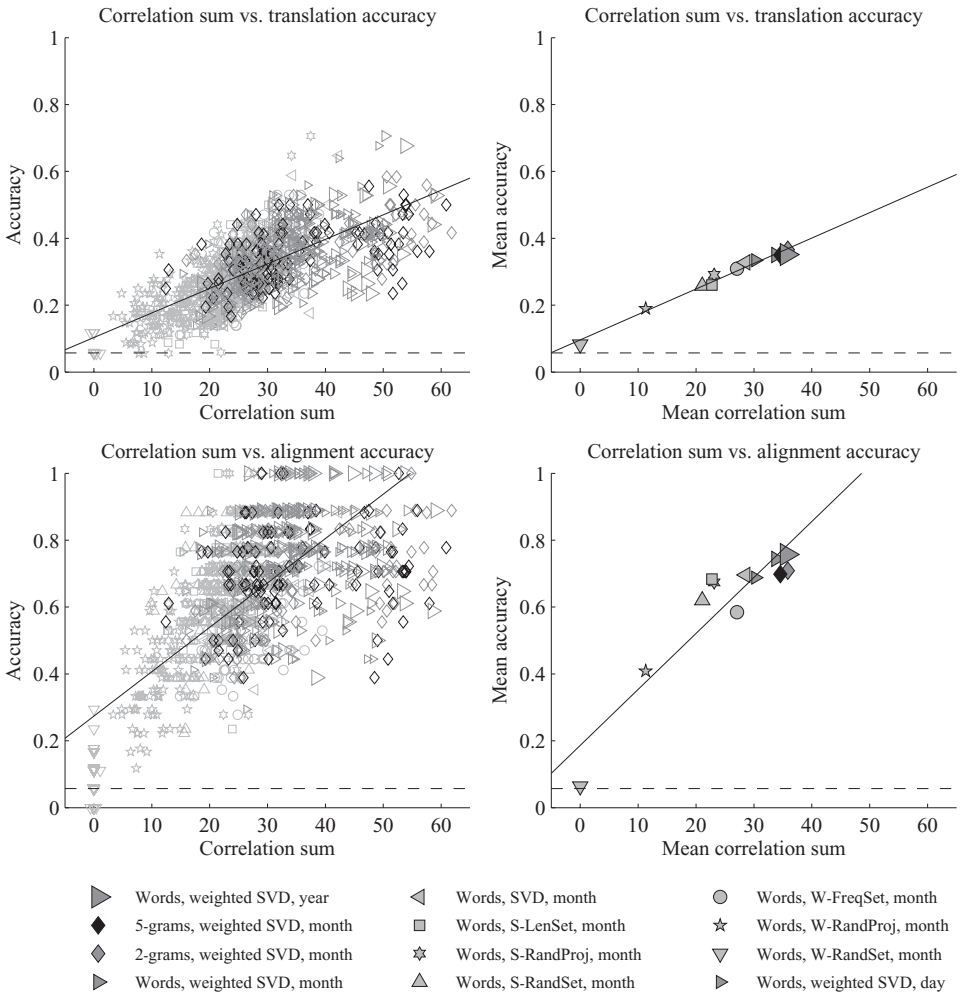

Fig. 17. Translation (top) and alignment (bottom) accuracy versus correlation sum over the test data subsets. The left side shows individual values for each subset and the right side shows the mean values for the different representations. The solid lines show the best linear least squares fit of data points. The dashed horizontal line shows the average baseline result of a random choice.

was 0.633 over the subsets and 0.888 over the means. We also considered how the two related tasks, translation and alignment, could predict each other as indirect evaluations. That is, we compared the translation and alignment accuracy based on the two tasks. The correlation coefficient was 0.627 over the subsets and 0.888 over the means. Hence, indirect evaluations based on two very similar tasks could not predict each other's performance any better than the proposed direct evaluation. This shows that our evaluation method is at least as good as evaluation methods based on a sentence matching task.

Finally, we calculated the results for representations of different dimensionalities. Figure 18 compares the correlation sums and accuracies of weighted SVD with $D \in$

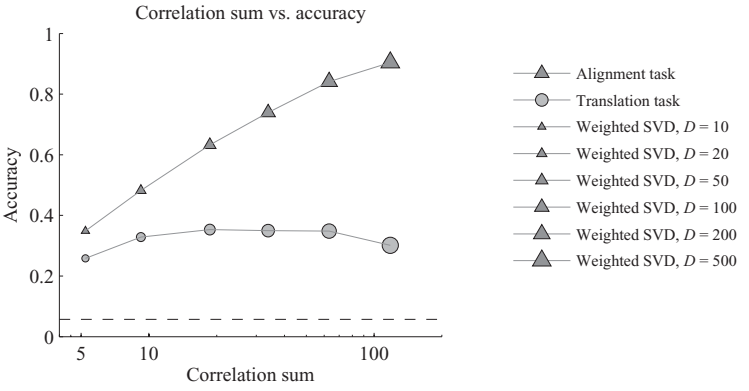

Fig. 18. Translation and alignment accuracies versus correlation sum. The points are mean values for 10, 20, 50, 100, 200, and 500 dimensional SVD representations.

$\{10, 20, 50, 100, 200, 500\}$ using the translation and alignment tasks. The alignment accuracy grows logarithmically with respect to the correlation sum, indicating that the components that are correlated only slightly are useful if weighted accordingly. In contrast, the translation accuracy starts to decrease already after $D = 50$. Thus, there seems to be no way to score the representations of different dimensionalities so that the scores would agree with all application evaluations.

5.4 Validation with manual evaluation of word translations

Our evaluation method based on CCA has an advantage that the canonical variates are interpretable: For a given canonical variate, we can find which original features, for example, words, contribute to it by calculating canonical factor loadings (6). As an example from SVD trained with one-month data, there is a variate that strongly correlates with English words *vote*, *place*, *take*, *tomorrow*, and *noon* and Finnish words *äänestys* (*vote* in English), *toimitetaan* (*takes place*), *klo* (*at* [*time*]), *huomenna* (*tomorrow*), and *12.00* (*noon*). While these are not direct translations, it is easy to see that they come from same sentences in each language. This is a desirable behavior due to the sentence-aligned corpus. We use the correctness of such word translations to compare different representations. A method performing well according to the evaluation method should also provide good translations.

We compared the five most positively and negatively correlated original features for both the languages to see if they refer to the same meaning. For the two sets of words corresponding to a given canonical variate, we counted the number of correct translations. Translations were checked in both the directions and the sum of the correct translations was used in the evaluation. Thus, the maximum score for one canonical variate is $(5 + 5) \times 2 = 20$. We computed the scores for each canonical variate and used the sum as an evaluation measure.

We used the one-month data with four different dimensionality reduction methods: SVD, S-LenSet, W-FreqSet, and W-RandProj with term frequency weighting and no normalization. In addition, two more settings were tested for SVD: entropy

Table 5. *Evaluation results of different dimensionality reduction methods and different settings for SVD using the correlation sum and the number of correct translations*

Method	Correlation sum (rank)	Correct translations (rank)
Weighted SVD, year data	41.5 (1.)	810 (1.)
Weighted SVD, month data	38.7 (2.)	787 (2.)
SVD	29.9 (3.)	642 (4.)
W-FreqSet	28.7 (4.)	773 (3.)
S-LenSet	24.3 (5.)	516 (5.)
W-RandProj	12.0 (6.)	421 (6.)

weighting together with L2 normalization for both the one-month and one-year data. The dimensionality was reduced to 100, which was then the number of canonical variates to evaluate.

Table 5 shows the correlation sum and the number of correct translations found for different vector representations. It can be clearly seen that the representations with high correlation sum also provided large number of correct translation pairs. Correlation between the two scores was 0.9772. The only difference in the ranks is for SVD and W-FreqSet. They reached almost the same correlation sum, but the evaluated factor loadings for W-FreqSet had more correct translations.

6 Discussion

In this work, we experimented with multilingual sentence-aligned data. We found that the results agreed generally well with those found in the literature, confirming that the proposed evaluation method is trustworthy. A natural follow-up is to test the evaluation method with longer pieces of text, and compare the results with a well-known task in, for example, IR. Especially the effect of the bag-of-phrases feature (Section 5.2.4), which was found to be more positive than expected, seems to be worth further study.

The experiment with different language pairs (Section 5.2.5) illustrates that the more related the language pair, the easier to get high evaluation scores. Due to more similar syntax and grammar, it is likely that the closely related languages have many correlated features that are not important for the semantic content. Thus, if we want to have as little non-semantic correlation as possible, then it may be a good idea to use two very different languages. However, if the languages are very different (consider, e.g., Chinese and English), then they are likely to need different feature extraction methods, making the evaluation setup more complicated.

The question on how the correlations found by the evaluation method should be combined to a single score is still partially open. If the number of dimensions in the evaluated representations is the same, the sum of correlations seems to work well. However, the validation with sentence matching tasks in Section 5.3 showed mixed results when the number of dimensions was varied. The score correlated well with the results of sentence alignment when we applied canonical variates weighted with the corresponding canonical correlations. However, the score did not correlate well

with the results of more difficult translation task, in which the variates were not weighted. It seems to be difficult to define a single measure that would work well with all applications.

The proposed evaluation method uses linear dependence calculation, which is very robust and fast. However, it may not always find the true dependence. One solution would be to use kernel CCA to extend the evaluation approach for nonlinear dependences. However, replacing linear CCA with kernel CCA in the evaluation has practical issues, which would make the evaluation setup more complicated. For example, several kernel functions would need to be tested, as the optimal one is likely to differ between the representations. Also, regularization would need to be optimized. Moreover, it is difficult to interpret the canonical variates in kernel space.

Another restriction of the evaluation method is that it works for monolingual feature extraction methods only. If the features were extracted from a bilingual corpus, high scores would only indicate that the discovered features are similar to those that CCA can find, rather than the most semantic ones. The evaluation setup actually encompasses cross-language feature extraction comparable to Vinokourov *et al.* (2003) or Zhang *et al.* (2010). The canonical variates and correlations found by CCA could be used to, for instance, identify words or phrases that do not contribute to the correlation in the common vector space and remove them from the representations.

While probabilistic models, such as cross-lingual PLSA by Zhang, Mei and Zhai (2010), can be applied to feature extraction similar to CCA, it is less clear if the models could be applied in an evaluation setup similar to ours. In topic models, such as PLSA, both the latent topics and observed features (usually words) are assumed to be multinomially distributed. This restricts the usage of PLSA to the evaluation of feature selection, that is, which words (or other discrete variables) to choose, and rules out all vector space models with continuous variables. Another practical problem with PLSA is the difficulty to decide the number of latent topics, as well as how to measure the amount of dependence or correlation between the original features giving the topics. Furthermore, PLSA models need to be inferred separately for each dimensionality, the results depend on the initialization, and the learning algorithms do not guarantee finding the global optimum.

The proposed evaluation setup could be extended to different types of correlated data sets, given that the desired features correspond to the information shared by the two sets. One possible direction is to consider a situation where the underlying features are not semantic. For example, if the target application is genre identification, the features typically include statistics on part-of-speech tags, word lengths, and punctuations (Finn and Kushmerick 2006). Given a data set where paired documents were of the same genre but otherwise different in content, a feature extraction that found genre-related features would get high correlations.

Another direction to extend the evaluation approach is to consider an asymmetric situation, where the correlated document sets and the applied feature extractions are of different type. For example, we could use full and summarized texts in the same language, or texts in professional and layperson language, as long as it is mainly the semantics that is shared by the data sets. Furthermore, if the first feature extraction

method is already known to give good semantic features for the respective data samples, it can be kept as it is, and only the second one is evaluated. There can also be a situation where the correlated data sets are of different modalities, for example, images and their descriptions. As the underlying semantic information is shared between data samples, it should be possible, although very challenging, to also evaluate feature extractions in this case.

7 Conclusions

We presented a novel idea of using canonical correlation analysis to evaluate the feature extraction methods that are used to construct vector representations of a language. The proposed evaluation method is simple, unsupervised, and language-independent. The only requirement is a bilingual corpus of paired samples, such as sentences or documents. The methods are evaluated by measuring the amount of cross-correlation between the extracted features of the paired samples. As the underlying semantic content of the samples can be assumed to be independent of the language, high correlations indicate that the evaluated method can capture much of the semantics. The evaluation approach was validated in various experiments using a sentence-aligned corpus. The evaluation results for bag-of-words representations agreed well with previous findings and with general intuition. The evaluation method was also able to find good vector representations for two different sentence matching tasks and for a task of finding word translations. These three sets of experiments showed that the proposed evaluation method provides a direct and reliable way to compare different vector representations.

References

Akaho, S. 2001. A kernel method for canonical correlation analysis. In *Proceedings of the International Meeting of the Psychometric Society (IMPS2001)*, Osaka, Japan. Berlin, Germany: Springer-Verlag.

Alpaydin, E. 2010. *Introduction to Machine Learning*, 2nd ed. Cambridge, MA, USA: MIT Press.

Bach, F. R., and Jordan, M. I. 2003. Kernel independent component analysis. *The Journal of Machine Learning Research* **3**: 1–48.

Bagga, A., and Baldwin, B. 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING'98)*, Montreal, Canada, vol. 1, pp. 79–85. New Brunswick, NJ, USA: Association for Computational Linguistics.

Benzécri, J.-P. 1973. *L'Analyse des Données. Vol. II. L'Analyse des Correspondances.* Paris, France: Dunod.

Bernard, J. R. L. (ed.). 1990. *The Macquarie Encyclopedic Thesaurus.* Sydney, Australia: The Macquarie Library.

Besançon, R., and Rajman, M. 2002. Evaluation of a vector space similarity measure in a multilingual framework. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, vol. 1252, Las Palmas, Spain. Paris, France: European Language Resources Association.

Bingham, E., and Mannila, H. 2001. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the 7th ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining (KDD-2001), San Francisco, CA, USA, pp. 245–250. New York, NY, USA: ACM.

Blei, D. M., Ng, A. Y., and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* **3**: 993–1022.

Borga, M. 1998. *Learning Multidimensional Signal Processing*. PhD thesis, Linköping University, Sweden.

Bradford, R. B. 2008. An empirical study of required dimensionality for large-scale latent semantic indexing applications. In *Proceeding of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*, Napa Valley, CA, USA, pp. 153–162. New York, NY, USA: ACM.

Caropreso, M. F., Matwin, S., and Sebastiani, F. 2001. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In A. G. Chin (ed.), *Text Databases & Document Management: Theory & Practice*, pp. 78–102. Hershey, PA, USA: IGI Publishing.

Chew, P., and Abdelali, A. 2007. Benefits of the 'massively parallel Rosetta stone': cross-language information retrieval with over 30 languages. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, Prague, Czech Republic, pp. 872–879. New Brunswick, NJ, USA: Association for Computational Linguistics.

Coenen, F., Leng, P., Sanderson, R., and Wang, Y. J. 2007. Statistical identification of key phrases for text classification. In *Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM '07)*, Leipzig, Germany, pp. 838–853. Berlin, Germany: Springer-Verlag.

Curran, J. R., and Moens, M. 2002. Scaling context space. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, Philadelphia, PA, USA, pp. 231–238. New Brunswick, NJ, USA: Association for Computational Linguistics.

De Bie, T., and De Moor, B. 2003. On the regularization of canonical correlation analysis. In *Proceedings of the Fourth International Symposium on Independent Component Analysis and Blind Source Separation (ICA2003)*, Nara, Japan, pp. 785–790. Kyoto, Japan: NTT Communication Science Laboratories.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science* **41**(6): 391–407.

Dumais, S. T. 1991. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments, & Computers* **23**(2): 229–236.

Fellbaum, C. (ed.). 1998. *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press.

Finn, A., and Kushmerick, N. 2006. Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology* **57**(11): 1506–1518.

Gaussier, É., Renders, J.-M., Matveeva, I., Goutte, C., and Déjean, H. 2004. A geometric view on bilingual lexicon extraction from comparable corpora. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain, vol. 4, pp. 526–533. East Stroudsburg, PA, USA: Association for Computational Linguistics.

Haghighi, A., Liang, P., Berg-Kirkpatrick, T., and Klein, D. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, Columbus, OH, USA, pp. 771–779. Stroudsburg, PA, USA: Association for Computational Linguistics.

Hardoon, D. R., and Shawe-Taylor, J. 2007. Sparse canonical correlation analysis. Technical Report, University College London, London, UK.

Hardoon, D. R., Szedmak, S., and Shawe-Taylor, J. 2004. Canonical correlation analysis: an overview with application to learning methods. *Neural Computation* **16**(12): 2639–2664.

Harman, H. H. 1960. *Modern Factor Analysis*. Chicago, IL, USA: University of Chicago Press.

Hofmann, T. 1999. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*, Stockholm, Sweden, pp. 289–296. San Francisco, CA, USA: Morgan Kaufmann.

Honkela, T., Hyvärinen, A., and Väyrynen, J. J. 2010. WordICA – emergence of linguistic representations for words by independent component analysis. *Natural Language Engineering* **16**: 277–308.

Hotelling, H. 1936. Relations between two sets of variates. *Biometrika* **28**(3): 321–377.

Johnson, W. B., and Lindenstrauss, J. 1984. Extensions of Lipschitz maps into a Hilbert space. *Contemporary Mathematics* **26**: 189–206.

Jones, K. S. 1972. A statistical interpretation of term specifity and its application in retrieval. *Journal of Documentation* **28**(1): 11–21.

Kanerva, P., Kristoferson, J., and Holst, A. 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society (CogSci 2000)*, Philadelphia, PA, USA, p. 1036. Mahwah, NJ, USA: Erlbaum.

Kaski, S. 1998. Dimensionality reduction by random mapping: fast similarity computation for clustering. In *Proceedings of International Joint Conference on Neural Networks (IJCNN'98)*, Anchorage, AK, USA, vol. 1, pp. 413–418. Piscataway, NJ, USA: IEEE.

Kay, J. 1992. Feature discovery under contextual supervision using mutual information. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN 1992)*, Baltimore, MD, USA, vol. 4, pp. 79–84. Los Alamitos, CA, USA: IEEE.

Kiss, G. R., Armstrong, C., Milroy, R., and Piper, J. 1973. An associative thesaurus of English and its computer analysis. In A. J. Aitkin, R. W. Bailey, and N. Hamilton-Smith (eds.), *The Computer and Literary Studies*, pp. 153–165. Edinburgh, UK: Edinburgh University Press.

Koehn, P. 2005. Europarl: a parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit*, Phuket, Thailand, pp. 79–86. Tokyo, Japan: Asia-Pacific Association for Machine Translation.

Koehn, P., Och, F. J., and Marcu, D. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL)*, Edmonton, Canada, pp. 48–54. Morristown, NJ, USA: Association for Computational Linguistics.

Koster, C. H. A., and Seutter, M. 2003. Taming wild phrases. In *Proceedings of the 25th European Conference on Information Retrieval Research (ECIR'03)*, Pisa, Italy, pp. 161–176. Berlin, Germany: Springer-Verlag.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2**: 83–97.

Lai, P. L., and Fyfe, C. 2000. Kernel and nonlinear canonical correlation analysis. *International Journal of Neural Systems* **10**(5): 365–377.

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

Leurgans, S. E., Moyeed, R. A., and Silverman, B. W. 1993. Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society. Series B (Methodological)* **55**(3): 725–740.

Lewis, D. D. 1992. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '92)*, Copenhagen, Denmark, pp. 37–50. New York, NY, USA: ACM.

Li, Y., and Shawe-Taylor, J. 2007. Advanced learning algorithms for cross-language patent retrieval and classification. *Information Processing and Management* **43**(5): 1183–1199.

Lund, K., and Burgess, C. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers* **28**(2): 203–208.

Manning, C. D., and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA, USA: MIT Press.

Manning, C. D., Raghavan, P., and Schütze, H. 2008. *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.

Melzer, T., Reiter, M., and Bischof, H. 2001. Nonlinear feature extraction using generalized canonical correlation analysis. In G. Dorffner, H. Bischof, and K. Hornik (eds.), *Proceedings of the International Conference on Artificial Neural Networks (ICANN '01)*, Vienna, Austria (vol. 2130 of *Lecture Notes in Computer Science*), pp. 353–360. Berlin, Germany: Springer-Verlag.

Mihalcea, R., and Simard, M. 2005. Parallel texts. *Natural Language Engineering* **11**(3): 239–246.

Minier, Z., Bodó, Z., and Csató, L. 2007. Wikipedia-based kernels for text categorization. In *Proceedings of the 9th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'07)*, Timisoara, Romania, pp. 157–164. Los Alamitos, CA, USA: IEEE Computer Society.

Mitchell, J., and Lapata, M. 2008. Vector-based models of semantic composition. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08:HLT)*, Columbus, OH, USA, pp. 236–244. Stroudsburg, PA, USA: Association for Computational Linguistics.

Nakov, P., Popova, A., and Mateev, P. 2001. Weight functions impact on LSA performance. In *Proceedings of the EuroConference on Recent Advances in Natural Language Processing (RANLP 2001)*, pp. 187–193. Tzigov Chark, Bulgaria: Bulgarian Academy of Sciences.

Nelson, D. L., McEvoy, C. L., and Schreiber, T. A. 1998. The University of South Florida word association, rhyme, and word fragment norms. `http://web.usf.edu/FreeAssociation/` Tampa, FL, USA: University of South Florida (Accessed 7 Oct 2010).

Rapp, R. 2002. The computation of word associations: comparing syntagmatic and paradigmatic approaches. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, Taipei, Taiwan, pp. 1–7, International Committee on Computational Linguistics. New Brunswick, NJ, USA: Association for Computational Linguistics.

Rapp, R. 2004. A freely available automatically generated thesaurus of related words. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, pp. 395–398. Paris, France: European Language Resources Association.

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

Roget, P. 1911. *Thesaurus of English Words and Phrases*. London, UK: Longmans, Green.

Rummel, R. J. 1970. *Applied Factor Analysis*. Evanston, IL, USA: Northwestern University Press.

Sadeniemi, M., Kettunen, K., Lindh-Knuutila, T., and Honkela, T. 2008. Complexity of European Union languages: a comparative approach. *Journal of Quantitative Linguistics* **15**(2): 185–211.

Sahlgren, M. 2006a. Towards pertinent evaluation methodologies for word-space models. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy. Paris, France: European Language Resources Association.

Sahlgren, M. 2006b. *The Word-Space Model*. PhD thesis, Department of Linguistics, Stockholm University, Stockholm, Sweden.

Sahlgren, M., and Karlgren, J. 2005. Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Natural Language Engineering* **11**(03): 327–341.

Salton, G. (ed.). 1971. *The SMART System – Experiments in Automatic Document Processing*. Upper Saddle River, NJ, USA: Prentice-Hall.

Salton, G., and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. *Information Prosessing and Management* **24**(5): 513–523.

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

Schütze, H. 1992. Dimensions of meaning. In *Proceedings of the 1992 ACM/IEEE Conference on Supercomputing (SC 1992)*, Minneapolis, MN, USA, pp. 787–796. Los Alamitos, CA, USA: IEEE Computer Society.

Schütze, H., and Pedersen, J. 1995. Information retrieval based on word senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval (SDAIR95)*, Las Vegas, NV, USA, pp. 161–175.

Schütze, H., Hull, D. A., and Pedersen, J. O. 1995. A comparison of classifiers and document representations for the routing problem. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '95)*, Seattle, WA, USA, pp. 229–237. New York, NY, USA: ACM.

Scott, S., and Matwin, S. 1999. Feature engineering for text classification. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML '99)*, Bled, Slovenia, pp. 379–388. San Francisco, CA, USA: Morgan Kaufmann.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys* **34**(1): 1–47.

Steyvers, M., Shiffrin, R. M., and Nelson, D. L. 2005. Word association spaces for predicting semantic similarity effects in episodic memory. In A. F. Healy (ed.), *Experimental Cognitive Psychology and Its Applications*, pp. 237–249. Washington, DC, USA: American Psychological Association.

Tripathi, A., Klami, A., and Kaski, S. 2008. Using dependencies to pair samples for multi-view learning. TKK Reports in Information and Computer Science TKK-ICS-R8, Helsinki University of Technology, Espoo, Finland.

Tripathi, A., Klami, A., and Virpioja, S. 2010. Bilingual sentence matching using kernel CCA. In *Proceedings of the 2010 IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2010)*, Kittilä, Finland, pp. 130–135. Los Alamitos, CA, USA: IEEE Press.

Turney, P. D. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In L. D. Raedt and P. A. Flach (eds.), *Proceedings of the Twelth European Conference on Machine Learning (ECML-2001)*, Freiburg, Germany (vol. 2167 of *Lecture Notes in Computer Science*), pp. 491–502. Berlin, Germany: Springer-Verlag.

Turney, P. D. 2005. Measuring semantic similarity by latent relational analysis. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)*, Edinburgh, UK, pp. 1136–1141. International Joint Conferences on Artificial Intelligence Organization. San Francisco, CA, USA: Morgan Kaufmann.

Väyrynen, J. J., Lindqvist, L., and Honkela, T. 2007. Sparse distributed representations for words with thresholded independent component analysis. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2007)*, Orlando, FL, USA, pp. 1031–1036. Piscataway, NJ, USA: IEEE.

Vinokourov, A., Shawe-Taylor, J., and Cristianini, N. 2003. Inferring a semantic representation of text via cross-language correlation analysis. *Advances in Neural Information Processing Systems* **15**: 1497–1504.

Yarowsky, D., and Florian, R. 2002. Evaluating sense disambiguation across diverse parameter spaces. *Natural Language Engineering* **8**(4): 293–310.

Zelikovitz, S., and Hirsh, H. 2001. Improving text classification with LSI using background knowledge. In Nebel, B. (ed.), *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI01)*, Seattle, WA, USA, pp. 113–118. International Joint Conferences on Artificial Intelligence Organization. San Francisco, CA, USA: Morgan Kaufmann.

Zesch, T., and Gurevych, I. 2009. Wisdom of crowds versus wisdom of linguists – measuring the semantic relatedness of words. *Natural Language Engineering* **16**(1): 25–59.

Zhang, D., Mei, Q., and Zhai, C. 2010. Cross-lingual latent topic extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, pp. 1128–1137. Stroudsburg, PA, USA: Association for Computational Linguistics.