Distributed Document and Phrase Co-embeddings for Descriptive Clustering

Motoki Sato[1], Austin J. Brockmeier[2], Georgios Kontonatsios[1], Tingting Mu[1], John Y. Goulermas[2], Jun'ichi Tsujii[3,1] and Sophia Ananiadou[1]

[1]University of Manchester, National Centre for Text Mining (NaCTeM), Manchester, UK
[2]Department of Computer Science, University of Liverpool, Liverpool, UK
[3]Artificial Intelligence Research Center, AIST, Tokyo, Japan
sato.motoki.sa7@is.naist.jp, a.j.brockmeier@liverpool.ac.uk
sophia.ananiadou@manchester.ac.uk

Abstract

Descriptive document clustering aims to automatically discover groups of semantically related documents and to assign a meaningful label to characterise the content of each cluster. In this paper, we present a descriptive clustering approach that employs a distributed representation model, namely the paragraph vector model, to capture semantic similarities between documents and phrases. The proposed method uses a joint representation of phrases and documents (i.e., a co-embedding) to automatically select a descriptive phrase that best represents each document cluster. We evaluate our method by comparing its performance to an existing state-of-the-art descriptive clustering method that also uses co-embedding but relies on a bag-of-words representation. Results obtained on benchmark datasets demonstrate that the paragraph vector-based method obtains superior performance over the existing approach in both identifying clusters and assigning appropriate descriptive labels to them.

1 Introduction

Document clustering is a well-established technique whose goal is to automatically organise a collection of documents into a number of semantically coherent groups. Descriptive document clustering goes a step further, in that each identified document cluster is automatically assigned a human-readable label (either a word or phrase) that characterises the semantic content of the documents within the cluster. Descriptive clustering methods have been shown to be useful in a variety of scenarios, including information retrieval (Bharambe and Kale, 2011), analysis of social networks (Zhao and Zhang, 2011), and large-scale exploration (Nassif and Hruschka, 2013) and visualisation of text collections (Kandel et al., 2012).

A number of previously proposed descriptive clustering techniques work by extending a standard document clustering approach. Documents are typically clustered based on a bag-of-words (BoW) representation (i.e., the occurrence counts of the words that appear in each document). Then, each cluster is labelled using the most commonly occurring word or phrase within the cluster (Weiss, 2006). In contrast to this approach, the recently proposed descriptive clustering approach (CEDL) (Mu et al., 2016) maps documents and candidate cluster labels into a common semantic vector space (i.e., co-embedding). The co-embedding space facilitates the straightforward assignment of descriptive labels to document clusters. The CEDL method has been shown to generate accurate cluster labels and achieved improved clustering performance when compared to standard descriptive clustering methods. Nonetheless, the co-embedding is based solely on a BoW representation of the documents and is thus limited in its ability to accurately represent the semantic similarity between documents.

In this paper, we investigate a specific case of descriptive clustering that selects a single multi-word phrase to characterise each cluster of documents (Li et al., 2008). Firstly, we assume descriptive phrases are to be selected from a candidate phrase set extracted from the corpus during preprocessing. The proposed method then follows the co-embedding descriptive clustering paradigm of the CEDL algorithm. However, in-

Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 991–1001,
Valencia, Spain, April 3-7, 2017. ©2017 Association for Computational Linguistics

stead of using a BoW representation, we employ the paragraph vector (PV) (Le and Mikolov, 2014) model to learn a distributed vector representations of phrases and documents. These distributed representations move beyond unstructured BoW representations by considering the local contexts in which words and phrases appear within documents, which provides a more precise estimate of semantic similarity.

In particular, we present two extensions to the initial PV-based method that enable models that learn a common co-embedding space of documents and phrases. The first extension jointly learns co-embeddings of documents and phrases. The second extension constructs 'pseudo-documents' consisting of the lexical context surrounding each occurrence of a particular phrase. Each of these contexts are treated as separate document instance that are associated with a single embedded vector. In both cases, after clustering the document embedding vectors, each embedded phrase is a candidate cluster label. To select the most appropriate descriptive label amongst these candidates, we first rank the documents according to their proximity to each candidate label's embedding vector and then select the phrase whose ranking maximises the average precision for a given cluster.

We compare the results obtained by our PV-based descriptive clustering method against two methods: spectral clustering (Shi and Malik, 2000), which only identifies clusters (but does not assign labels to them), and the previously introduced CEDL method (Mu et al., 2016), which carries out both clustering and labelling. Experimental results based on publicly available benchmark text collections demonstrate the effectiveness and superiority of our methods in both clustering performance and labelling quality.

2 Related Work

2.1 Descriptive Clustering

Descriptive clustering methods typically use an unsupervised approach to firstly group documents into flat or hierarchical clusters (Steinbach et al., 2000). Document clusters are then characterised using a set of informative and discriminative words (Zhu et al., 2006), phrases (Mu et al., 2016; Li et al., 2008) or sentences (Kim et al., 2015).

Early approaches to descriptive clustering followed the description-comes-first (DCF) paradigm (Osiński et al., 2004; Weiss, 2006; Zhang, 2009). DCF-based methods work by firstly identifying a set of cluster labels, and subsequently forming document clusters by measuring the relevance of each document to a potential cluster label. DCF-based approaches have several shortcomings, which include poor clustering performance and low readability of cluster descriptors (Lee et al., 2008; Carpineto et al., 2009).

More recent developments in descriptive clustering have proposed alternative techniques, which approach the problems of improving clustering performance and descriptive label quality from various different angles. For instance, Scaiella et al. (2012) identifies Wikipedia concepts in documents and then computes relatedness between documents according to the linked structure of Wikipedia. Navigli and Crisafulli (2010) propose a method that takes into account synonymy and polysemy. Their method utilises the Google Web1T corpus to identify word senses based on word co-occurrences and computes the similarity between documents using the extracted sense information.

More recently, Mu et al. (2016) presented their co-embedding based descriptive clustering approach that learns a common co-embedding vector space of documents and candidate descriptive phrases. The co-embedded space simplifies the clustering and cluster labelling task into a more straightforward process of computing similarity between pairs of documents and between documents and candidate cluster labels.

2.2 Distributed Representation

Distributed representation techniques are becoming increasingly important in a number of supervised learning tasks, e.g., sentiment analysis (Dai et al., 2015), text classification (Dai et al., 2015; Ma et al., 2015) and named entity recognition (Turian et al., 2010). A number of models have been proposed to learn distributed word or phrase representations in order to predict word occurrences given a local context (Mnih and Hinton, 2009; Mikolov et al., 2013b; Mikolov et al., 2013a; Pennington et al., 2014). Subsequently, the PV model was proposed to learn representations of both words and documents (Le and Mikolov, 2014; Dai et al., 2015). The PV model has been shown to be capable of learning a semantically richer representation of documents compared to unstructured BoW models. To our knowledge, our work constitutes the first attempt to use distributed representation models to co-embed documents and phrases for unsupervised descriptive clustering.

3 Proposed Descriptive Clustering Method

As outlined above, the descriptive clustering task (i.e., grouping documents according to semantic relatedness and characterising the cluster content using a representative descriptive phrase) relies heavily on learning a representation of documents and phrases that can accurately capture relevant semantic information. A particularly effective strategy for descriptive clustering is to jointly map documents and descriptive phrases together into a common embedding space (Mu et al., 2016). The clustering of documents and selection of descriptive phrases for each cluster is then carried out by calculating the cosine similarities between documents (to form clusters), and between documents and descriptive phrases (to determine descriptive labels) in the learned space. Instead of relying on the commonly used BoW model, we propose a novel descriptive clustering approach. Our method uses similarities computed from distributed joint embeddings of documents and phrases, which are learned by considering both the global context provided by the document and the local context of the descriptive phrases. We propose two different strategies to learn these embeddings, as described below.

3.1 Joint Learning of Document and Phrase Embeddings

The first strategy jointly learns the distributed representations for documents and phrases by representing phrases, words, and documents as vectors that are used both to predict the occurrence of words in given documents (reflecting global document content information), and to predict the co-occurrences of words and phrases within a sliding window, to reflect the local context information.

We extend the PV model described in Dai et al. (2015) to simultaneously generate word, phrase and document embeddings. The objective function is to maximise the log probability of words and phrases conditioned on either their global or local context:

$$\sum_{t \in \mathcal{T}_P} \log \mathrm{p}(p_t|d_t) + \frac{1}{|\mathcal{C}_t|} \sum_{c \in \mathcal{C}_t} \log \mathrm{p}(p_t|c) \qquad (1)$$
$$+ \sum_{s \in \mathcal{T}_W} \log \mathrm{p}(w_s|d_s) + \frac{1}{|\mathcal{C}_s|} \sum_{c \in \mathcal{C}_s} \log \mathrm{p}(w_s|c)$$

where $\mathcal{T}_\mathcal{P}$ is the set of training phrase instances; $p_t \in \mathcal{P}$ is the t-th phrase instance; d_t denotes the document corresponding to the t-th training instance; c denotes a member of the local context $\mathcal{C}_t = [q_{t-L}, \ldots, q_{t-1}, q_{t+1}, \ldots, q_{t+L}]$, which occurs within a window size of L of the training instance ($|\mathcal{C}_t| = 2L$) and consists of both words and phrases $q_t \in \mathcal{P} \cup \mathcal{W}$; likewise, $\mathcal{T}_\mathcal{W}$ is the set of training word instances; $w_s \in \mathcal{W}$ is the s-th target word instance; d_s denotes the document corresponding to the s-th training instance; and $\mathcal{C}_s$ is its local context with $|\mathcal{C}_s| = 2L$. To summarise, the probability terms $\mathrm{p}(p_t|d_t)$ and $\mathrm{p}(w_s|d_t)$ model the document content information from a global level, while $\mathrm{p}(p_t|c)$ and $\mathrm{p}(w_s|c)$ model the local context. There are $2L + 1$ conditional probabilities estimated for each training instance.

The probability of a given lexical unit $q_t \in \mathcal{P} \cup \mathcal{W}$ (either a word or a phrase) is modelled using the vector embeddings of the $|\mathcal{P}|+|\mathcal{W}|$ words and phrases and the softmax function as follows:

$$\mathrm{p}(q_t|d_t) = \frac{\exp(\mathbf{u}_{q_t}^\top \mathbf{z}_{d_t})}{\sum_{q \in \mathcal{P} \cup \mathcal{W}} \exp(\mathbf{u}_q^\top \mathbf{z}_{d_t})} \qquad (2)$$

$$\mathrm{p}(q_t|c) = \frac{\exp(\mathbf{u}_{q_t}^\top \mathbf{z}_c)}{\sum_{q \in \mathcal{P} \cup \mathcal{W}} \exp(\mathbf{u}_q^\top \mathbf{z}_c)} \qquad (3)$$

where $\mathbf{u}_{q_t}$ is a weight vector specific to the target word or phrase, $\mathbf{z}_{d_t}$ is the embedding vector of the document corresponding to instance t, and $\mathbf{z}_c$ is the embedding vector of a word or phrase in the context of q_t. Since the document, phrase and word vectors all use the same weight vector $\mathbf{u}_{q_t}$ to predict the target phrase, they are necessarily in the same vector space.

3.2 Phrase Embeddings via Local Context Pseudo-Documents

The previous model considers learning an embedding as a multi-objective problem by trying to predict phrases and words based on the global and local context. Besides indexing, Equation (1) treats words and descriptive phrases interchangeably. An alternative approach is to treat phrases as ‘pseudo-documents’ by using the sets of words appearing in the local context of each phrase occurence. Specifically, training instances for a phrase’s embedding vector are constructed by extracting the local context around each occurrence of a phrase in the document collection. Using the augmented training set, consisting of both the original documents and the additional pseudo-documents, we can then employ any existing PV model (Le and Mikolov, 2014; Dai et al., 2015) to learn the document-phrase co-embeddings.

However, due to the significant differences in the sizes and numbers of documents and pseudo-documents, there is a danger that the addition of the pseudo-documents can have a detrimental effect on the performance of the model. Thus, we adopt a two-stage training procedure. Firstly, an embedding model is trained using only the documents. Then, we fix the weights of the model and optimise the phrase embeddings by providing the pseudo-documents as the input to the model.

We have integrated the above-mentioned process with two PV approaches, namely the distributed memory model (PV-DM) Le and Mikolov (2014), and the extension of the distributed BoW model (PV-DBOW) in Dai et al. (2015).

In the PV-DM model, the probability that a target word will appear in a given lexical context is conditioned on the surrounding co-occurring words and also the document:

$$\sum_{t \in \mathcal{T}_\mathcal{W}} \log \mathrm{p}(w_t | \mathcal{C}_t, d_t), \tag{4}$$

where w_t is the target word for instance t, $\mathcal{T}_\mathcal{W}$ is the set of training word instances, $\mathcal{C}_t = [w_{t-L}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+L}]$ are context words that occur within a window size of L words around w_t, and d_t denotes the document corresponding to the t-th training instance. The probability is modelled using a softmax function.

For phrase p, the objective is to maximise the sum of the log probabilities $\sum_{t \in \mathcal{T}_p} \log \mathrm{p}(w_t | \mathcal{C}_t, p)$ where $w_t \in \mathcal{T}_p$ are the word instances that appear in local context around the phrase, i.e., $\mathcal{T}_p$ is the set of word instances across all pseudo-documents, and $\mathcal{C}_t$ is the set of words that occur around the t-th word instance which also occur within the pseudo-documents for the phrase. Explicitly, the optimal embedding vector for the phrase is determined by solving the following optimisation problem:

$$\max_{\mathbf{z}_p} \sum_{t \in \mathcal{T}_p} \log \frac{\exp(\mathbf{u}_{w_t}^\top \mathbf{x}_t + \mathbf{v}_{w_t}^\top \mathbf{z}_p)}{\sum_w \exp(\mathbf{u}_w^\top \mathbf{x}_t + \mathbf{v}_w^\top \mathbf{z}_p)} \tag{5}$$

$$\mathbf{x}_t = [\mathbf{x}_{w_{t-L}}^\top, \ldots, \mathbf{x}_{w_{t-1}}^\top, \mathbf{x}_{w_{t+1}}^\top, \ldots, \mathbf{x}_{w_{t+L}}^\top]^\top$$

where $\mathbf{x}_t$ is the concatenation of all word vectors in the context of word w and $\{\mathbf{u}_w\}_w$ and $\{\mathbf{v}_w\}_w$ for w. To find an approximate solution, the parameters of the embedding vector are randomly initialised and optimised using stochastic gradient descent; the gradient is calculated via backpropagation (Rumelhart et al., 1986).

The PV-DBOW model simplifies the PV-DM model by ignoring the local context of words in the log probability function. The probability that a target word will appear in a given lexical context is conditioned solely by the document. Dai et al. (2015) introduced a modified version of the PV-DBOW model that treats words and documents as interchangeable inputs to the neural network. This enables the model to jointly learn word and document embeddings in the same space; we denote the model as PV-DBOW-W. Essentially, the objective of the PV-DBOW-W model is a combination of both the skip-gram model (Mikolov et al., 2013b) that generates word embeddings and the PV-DBOW method which is used for learning document embeddings:

$$\sum_{t \in \mathcal{T}_W} \log \mathrm{p}(w_t | d_t) + \frac{1}{|\mathcal{C}_t|} \sum_{c \in \mathcal{C}_t} \log \mathrm{p}(w_t | c). \tag{6}$$

To optimise the embedding of a specific phrase, denoted p, the existing word embeddings remain fixed, and the objective function is simplified as $\sum_{t \in \mathcal{T}_p} \log \mathrm{p}(w_t | p)$ where $w_t \in \mathcal{T}_p$ are word instances that appear in the local contexts around the phrase. The optimal embedding vector for this phrase is determined by solving the following optimisation problem:

$$\max_{\mathbf{z}_p} \sum_{t \in \mathcal{T}_p} \log \frac{\exp(\mathbf{u}_{w_t}^\top \mathbf{z}_p)}{\sum_w \exp(\mathbf{u}_w^\top \mathbf{z}_p)} \tag{7}$$

where the weight vectors $\{\mathbf{u}_w\}_w$ are fixed. As in the previous model, the parameters of the embedding vector are randomly initialised and optimised using stochastic gradient descent.

3.3 Descriptive Phrase Selection

Given co-embeddings of documents and phrases, any clustering algorithm can be applied. We use k-means, with the cosine similarity-based distance metric, to cluster the documents. Given the set of documents within each identified cluster $\mathcal{G}_1, \ldots, \mathcal{G}_K$, the document embedding vectors $\{\mathbf{z}_d\}_{d=1}^{N}$ and the descriptive phrase embedding vectors $\{\mathbf{z}_p\}_{p=1}^{P}$, we then select a descriptive phrase that best represents the documents assigned to a cluster.

A baseline approach for descriptive phrase selection is to select the phrase whose embedding vector is nearest to the cluster centroid; however, proximity to the cluster centroid is not always a good indicator of cluster membership, as it ignores the location of documents belonging to other clusters. An ideal phrase vector should lie closer to documents within the cluster than documents outside of the cluster. Accordingly, we rank documents based on their proximity to a candidate phrase and calculate the average precision of this ranking (where documents belonging to the given cluster are the true positives).

For cluster $\mathcal{G}$, we define the cluster membership indicator for each document as:

$$y_d = \begin{cases} 1 & d \in \mathcal{G} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

For a given phrase p, let $\pi_p(1)$ be the index of the nearest document to the phrase, and $\pi_p(i)$ be the index of the i-th nearest neighbour. The precision after the k-nearest documents are retrieved is $P_{\pi_p}(k) = \frac{1}{k}\sum_{i=1}^{k} y_{\pi_p(i)}$. The phrase which maximises the average precision $\overline{P}_p$ is selected as the cluster descriptor

$$p^* = \arg\max_{p} \left\{ \overline{P}_p = \frac{1}{|\mathcal{G}|} \sum_{k \in |\mathcal{G}|} P_{\pi_p}(k) \right\}, \tag{9}$$

where $|\mathcal{G}|$ is the number of documents in the cluster.

4 Results

We evaluate the proposed PV-based descriptive clustering methods in terms of cluster quality and descriptive phrase selection. Additionally, we show a visualisation of the co-embedding space in the supplementary material.

4.1 Datasets

We use two well-known, publicly available datasets: "Reuters-21578 Text Categorization Test Collection" from the Reuters newswire (Lewis, 1997), and the "20 Newsgroups" email dataset[1]. We pre-process the 20 Newsgroups corpus to remove email header information while for both datasets we extract candidate phrases using Termine (Frantzi et al., 2000), an automatic term extraction tool.

For the Reuters corpus, we use the complete document collection for training the PV models. For evaluation, we use both the training and testing sets of the modApte split, and select the 10 categories with the largest number of documents. Moreover, we remove documents that belong to multiple categories, this process results in an evaluation set of $8,009$ documents. For the 20 Newsgroups dataset, we use the complete set of $18,846$ documents for training the PV models. We remove words and phrases that only appear in a single document and then remove any empty documents. This process results in an evaluation set of $18,813$ documents with 20 categories, organised into 4 higher level parent categories. Table 1 summarises various characteristics of the employed datasets, including: a) number of documents, b) number of candidate phrases and c) category labels.

Table 1: Categories included in the evaluation subsets. 'R10' corresponds to the 10 largest categories after removing documents with multiple categories; the number of documents is in parentheses. All 20-Newsgroups categories have between 628 and 997 documents.

Reuters - 8,009 docs - 9,984 words - 11,732 phrases	
R10	earn(3923), acq(2292), crude(374), trade(327), money-fx(293), interest(271), money-supply(151), ship(144), sugar(122), coffee(112)
20 News - 18,813 docs - 43,285 words - 36,041 phrases	
sci	crypt, electronics, med, space
comp	os.ms-windows.misc, sys.ibm.pc.hardware, graphics, windows.x, sys.mac.hardware
rec	autos, motorcycles, sport.baseball, sport.hockey
mix	comp.os.ms-windows.misc, rec.autos, rec.sport.baseball, sci.med, sci.space
all	*

4.2 Paragraph Vector Models

In this section, we provide implementation details for the three PV models (PV-DBOW-WP,

[1] http://qwone.com/~jason/20Newsgroups/

PV-DBOW-W, and PV-DM), introduce a fourth model (PV-CAT) and explain the different settings that we use throughout the experiments. The PV-DBOW-WP model is used to jointly train phrase, word and document co-embeddings. For the PV-DBOW-W and PV-DM models, we use the two-stage training approach, in which the document embeddings and softmax weights are trained first, and then the phrase co-embeddings are trained using pseudo-documents. A window size of 10 words around the target phrase is used as the local context to create the pseudo-documents.

Each PV model has a number of parameters, including the dimension of the embedded spaces and the size of the context window. We set all embedding dimensions to 100. For the PV-DBOW-W and PV-DBOW-WP model, we use a context window of 10 words/phrases while for the PV-DM model a window size of 2 words (we tuned the size of the context window by applying the two PV models to a small development set of the Reuters corpus). This disparity in window size is not surprising since the PV-DM model considers the order of words within the local context and uses different parameters for the vectors at each location in the context window, Equation (5), whereas an increased window size does not add additional parameters to the PD-DBOW model.

We create an additional model, namely PV-CAT, by concatenating the vector representations induced by the PV-DBOW-W and the PV-DM models. This is performed after training the document and the phrase vectors. Intuitively, the concatenation of the PV-DBOW-W and PV-DM feature vectors can provide complimentary information given that the two models are trained using a different size of context window (i.e., 10 and 2 words, respectively).

Given that the size of the vocabulary is very large, computing the softmax function during stochastic gradient descent is computationally expensive. For faster training, different optimisation algorithms can be used to approximate the log probability function. We use a combination of negative sampling and hierachical softmax via backpropagation (Mnih and Hinton, 2009; Mikolov et al., 2013b). Specifically, we use negative sampling and then further optimise the embeddings using hierarchical softmax. Although, these are different optimisation approaches, both methods can be applied in this ad-hoc manner. Moreover, we follow the process described in Le and Mikolov (2014) to tune the learning rate. For this, we set the initial learning rate to 0.025 and decrease it linearly during 10 training epochs such that the learning rate is 0.001 during the last training epoch.

4.3 Baseline Methods

As our first baseline, we perform spectral clustering based on the affinity matrix produced according to the cosine similarity between the standard term-frequency inverse document (tf-idf) representation of the documents. We used the normalised cut (NC) spectral clustering algorithm proposed by Shi and Malik (2000).

We also compare our proposed method to the CEDL algorithm (Mu et al., 2016), which uses a measure of second-order similarity between phrases and documents, based on their co-occurrences at the document level, to obtain a spectral co-embedding. We use the same parameters suggested in the original publication, but carried out minor changes to the algorithm to allow the method to be scaled up to larger datasets. To compare clustering performance, we also run the CEDL algorithm without the phrase co-embeddings.

4.4 Evaluation of Cluster Quality

In this experiment, we evaluate the clustering performance of the methods by comparing automatically generated document clusters against the gold standard categories. For all methods, we use k-means clustering with cosine similarity as the distance metric. Following previous approaches (Xie and Xing, 2013), we set the number of clusters equal to the number of gold standard categories. As evaluation metrics, we use the macro-averaged F1 score[2], and normalised mutual information[3].

Table 2 compares the clustering performance achieved by four PV models (PV-DBOW-WP, PV-DBOW-W, PV-DM, and PV-CAT) against the performance of the baselines (i.e., the two versions of the CEDL algorithm and spectral clustering via normalised cut).

[2] For each category, the maximum F1 score across the clusters is used.

[3] Normalised mutual information $\frac{MI(G,C)}{\max\{H(G),H(C)\}}$ is defined as the mutual information between the automatically generated clusters and gold standard categories $MI(G, C)$ divided by the maximum of the entropy of the clusters $H(G)$ or the categories $H(C)$.

Table 2: Clustering performance assessed by correspondence measures to gold standard categories. *NC*: spectral clustering via normalised cut, *CE1*: CEDL, *CE2*: CEDL without phrase co-embeddings, *PV1*: PV-DBOW-WP, *PV2*: PV-DBOW-W, *PV3*: PV-DM, *PV4*: PV-CAT.

	NC	*CE1*	*CE2*	*PV1*	*PV2*	*PV3*	*PV4*
F1 (macro-averaged; higher is better)							
R10	0.60	0.54	0.56	0.54	**0.66**	0.48	**0.66**
sci	0.89	0.89	0.88	0.91	0.91	0.90	**0.92**
comp	0.48	0.46	0.55	**0.67**	0.66	0.63	0.65
rec	0.87	0.86	0.90	**0.92**	**0.92**	0.88	**0.92**
mix	0.93	0.92	0.93	**0.95**	0.94	0.93	**0.95**
all	0.56	—	—	**0.69**	**0.69**	0.66	**0.69**
all†	0.52	0.48	0.55	**0.67**	**0.67**	0.64	**0.67**
Normalised mutual information (higher is better)							
R10	0.42	0.41	0.42	0.47	0.50	0.40	**0.51**
sci	0.68	0.68	0.66	0.72	0.71	0.70	**0.74**
comp	0.25	0.22	0.30	**0.40**	0.39	0.36	0.39
rec	0.69	0.67	0.74	**0.78**	**0.78**	0.70	**0.78**
mix	0.80	0.77	0.79	**0.84**	0.82	0.80	0.83
all	0.51	—	—	0.62	0.62	0.60	**0.63**
all†	0.50	0.44	0.52	0.63	0.62	0.60	**0.64**

† Average of 10 random subsets using 10% of each category's documents.

The PV-DBOW-W and PV-CAT methods yield the best clustering performance on the Reuters dataset. Performance gains over the three baseline methods range between $6\% - 12\%$ (F1 score) and $8\% - 9\%$ (normalised mutual information). On the 20 Newsgroups dataset, the PV-DBOW-WP and PV-CAT models outperformed the baseline methods[4] by approximately $2\% - 21\%$ (F1 score) and $3\% - 18\%$ (normalised mutual information). Moreover, we note that the performance achieved by the PV-CAT model exceeds the best results reported in Xie and Xing (2013) (normalised mutual information of 0.6159 using a multi-grain clustering topic model).

Finally, we note that co-embedding is not designed as a mechanism for improving cluster quality. For CEDL, the co-embedding of phrases reduced the clustering performances in most datasets. This reduction in performance is equally observed for the paragraph vector models when applied to the Reuters dataset: the jointly trained co-embedding model (i.e., PV-DBOW-WP) achieved a lower performance than the two-stage approach PV-DBOW-W (F1 score of 0.56 and 0.66, respectively).

4.5 Evaluation of Cluster Labelling

In this section, we evaluate the cluster labels (i.e., multi-word phrases) selected by the proposed PV-based descriptive clustering methods. As a baseline approach, we use the CEDL algorithm that produces a co-embedded space of documents and phrases[5].

For each document cluster, we apply the phrase selection criterion, Equation (9), to identify the phrase that best describes the underlying cluster. Then for each gold standard category, the cluster having the highest proportion of documents belonging to the category is determined. This process means some clusters are assigned to multiple categories while other clusters are left unassigned. For each assigned cluster, we rank all documents according to their similarity to the automatically selected phrase (in the co-embedding space), where documents within the cluster have precedence over documents outside the cluster. We evaluate the quality of the cluster label by computing the average precision of this ranking in recalling the gold standard category. The average precision is maximised when documents closest to the selected phrase belong to the gold standard category. Table 3 shows the selected cluster descriptors aligned to the gold standard categories, the average precision and mean average precision scores achieved by the CEDL method and PV-based descriptive clustering models when applied to the Reuters and the 20 Newsgroups dataset.

The Reuters dataset presents a challenging case for descriptive clustering methods given that the distribution of gold standard categories is highly skewed, i.e., the majority categories (e.g., 'earn' and 'acq') correspond to more than one clusters while the remaining clusters cover multiple smaller categories. Nonetheless, we observe that the automatically selected cluster descriptors are related to the corresponding gold standard categories (e.g., 'import coffee' and 'oil export' for gold standard category 'ship'). In practice, the skewed distribution of gold standard categories can be addressed by using a larger number of clusters in k-means, or by using a cluster algorithm more amenable to heterogeneously sized clusters.

The 20 Newsgroups dataset shows a more balanced distribution of categories than the Reuters

[4] Our implementation of the CEDL algorithm did not scale up to the entire dataset ('all'), but average results on random subsets were consistent, as shown in the table.

[5] The spectral clustering via normalised cut and the CEDL algorithm without document and phrase co-embeddings are document clustering methods but not descriptive clustering models and thus they are excluded from this experiment.

Table 3: Cluster descriptions and average precision (as percentages) achieved by descriptive clustering methods. *CE1*: CEDL, *PV1*: PV-DBOW-WP, *PV2*: PV-DBOW-W, *PV3*: PV-DM, *PV4*: PV-CAT. The average precision metric depends on not only the phrase but also the location of documents relative to the selected phrase; consequently, the average precision of a phrase may vary among the embeddings.

Category	*CE1*		*PV1*		*PV2*		*PV3*		*PV4*	
earn	memotec datum	85	payable april	76	mln oper rev	89	mth jan	77	mln oper rev	88
acq	undisclosed sum	80	tender offer	58	undisclosed sum	73	tender offer	70	definitive agreement	70
crude	opec oil	92	venezuelan crude oil	47	oil production	88	oil export	79	oil production	91
trade	european community	49	trade relation	48	trade problem	75	representative clayton	77	trade problem	79
money-fx	bank rate	18	commercial lending rate	13	trade problem	17	deposit rate	22	trade problem	16
interest	bank rate	68	commercial lending rate	36	week t	49	deposit rate	53	week t	45
ship	european community	8	import coffee	7	raw sugar	15	oil export	11	costa rica	14
sugar	european community	67	import coffee	17	raw sugar	79	representative clayton	8	costa rica	34
money-supply	bank rate	9	commercial lending rate	15	week t	33	deposit rate	21	week t	37
coffee	european community	8	import coffee	43	raw sugar	25	representative clayton	8	costa rica	68
Unused clusters:	furman selz report		improve earning		share takeover offer		definitive agreement		share takeover offer	
	loss nil		undisclosed term		high earning		april record		revenue growth	
	bid null		group turnover		tax profit		exclude loss		april record	
	record today		current qtr		april record		mln net		mln dividend	
	present intention						net shr profit			
Mean average precision		48		36		54		42		54
sci.crypt	clipper key	93	key registration	94	back door	95	encryption method	95	back door	96
sci.electronics	transistor circuit	87	power amp	90	voltage divider	89	radio shack	86	circuit board	90
sci.med	other doctor	93	tech people	95	other treatment	94	other symptom	94	other treatment	95
sci.space	earth orbit	94	deep space	95	first spacecraft	95	low earth orbit	94	low earth orbit	96
comp.os.ms-windows.misc	trident 8900c	24	enhance mode	62	ms speaker sound driver	60	window version	55	dos app	59
comp.graphics	image file	44	art scene	70	art scene	74	swim chip	68	facet based modeller	70
comp.sys.ibm.pc.hardware	scsi hard drive	47	ide drive	60	tape drive	51	cmos setup	55	tape drive	52
comp.windows.x	application code	56	other widget	85	return value	77	widget name	79	return value	78
comp.sys.mac.hardware	apple price	59	mac lc	69	apple price	63	extra box	47	apple price	63
rec.autos	auto car	92	luxury sedan	94	same car	92	new car	89	sport car	94
rec.motorcycles	other bike	95	cruiser rider	95	same site	94	dod ama icoa nia	89	waterski bike	95
rec.sport.baseball	padded bat	88	leadoff hitter	94	playoff team	95	total baseball	91	playoff team	95
rec.sport.hockey	hockey playoff	92	cup final	94	nhl results	96	cup final	90	cup final	95
comp.os.ms-windows.misc	dos window font	95	auto show	97	dos window	97	dos window	97	dos window	98
rec.autos	bmw car	96	other car	97	same car	96	new car	97	other car	97
rec.sport.baseball	baseball fan	98	worst team	99	more game	98	last season	98	baseball season	99
sci.med	other doctor	94	other medical problem	96	many patient	94	other symptom	93	many patient	95
sci.space	earth orbit	94	japanese space agency	95	solar power	94	lunar surface	94	solar power	96
Mean average precision		80		88		86		84		87

corpus, and we note that all descriptive clustering methods were able to identify meaningful cluster descriptors that have a clear correspondence to the gold standard categories (e.g., 'window version' and 'dos app' for the category 'comp.os.ms-windows.misc').

With regard to the mean average precision, we observe that the PV-DBOW-W and PV-CAT models obtained the best performance. Moreover, the PV-CAT model achieved statistically significant improvements over the CEDL baseline in terms of the average precision across the 28 categories while no significant[6] improvement was observed for the remaining three PV-based models.

The results that we obtained demonstrate that the PV-based co-embedding space can effectively capture semantic similarities between documents and phrases. An illustrative example of this is shown in Table 4. In this example, we selected two documents that neighbour the phrase "user interface" in the PV-CAT co-embedded feature space for the "20 Newsgroups" dataset. It can be noted that although neither of the two documents explicitly contain the input phrase, the first discusses a semantically similar topic, and the second uses the acronym GUI, i.e., graphical user interface.

As another example, we generate a two-dimensional visualisation of the document-phrase co-embeddings using t-SNE (van der Maaten and Hinton, 2008) that demonstrates how co-embedded phrases can be used as 'landmarks' for exploring a corpus. For this example, we use the 'sci' categories from the 20 Newsgroup corpus and select the 200 most frequent phrases in this subset. As input to t-SNE, we use the chordal distance defined by the cosine similarity in the co-embedding space and set the perplexity level to 40. Figure 1 in the supplementary material shows the visualisation with the cluster boundaries, location of the documents and co-embedded phrases.

5 Conclusion

Descriptive document clustering helps information retrieval tasks by automatically organising document collections into semantically coherent groups and assigning descriptive labels to each

[6] For significance testing, we used a paired sign-test, with a significance threshold of 0.05 and Bonferroni multiple test correction for the 4 tests; the uncorrected p-value for the PV-CAT model is 0.0009.

Table 4: Two documents whose vector embeddings were the 5th and 6th nearest neighbours (according to the cosine of the angle of the corresponding vectors) to the phrase “user interface” in the PV-CAT based co-embedded space.

`train/comp.windows.x_67337` *Does anyone know the difference between MOOLIT and OLIT? Does Sun support MOOLIT? Is MOOLIT available on Sparcstations? MoOLIT (Motif/Open Look Intrinsic Toolkit allows developers to build applications that can switch between Motif and Open Look at run-time, while OLIT only gives you Open Look.* *Internet: chunhong@vnet.ibm.com*
`test/comp.windows.x_68238` *Hi there,* *I’m looking for tools that can make X programming easy. I would like to have a tool that will enable to create X motif* ***GUI*** *Interactivly. Currently I’m Working on a SGI with forms. A package that enables to create* ***GUI*** *with no coding at all (but the callbacks).* *Any help will be appreciated.* *Thanks Gabi.*

group. In this paper, we have presented a descriptive clustering method that uses paragraph vector models to support accurate clustering of documents and selection of meaningful and precise cluster descriptors. Our PV-based approach maps phrases and documents to a common feature space to enable the straightforward assignment of descriptive phrases to clusters. We have compared our approach to another state-of-the-art algorithm employing a co-embedding based on bag-of-word representations. The PV-based descriptive clustering method achieved superior clustering performance on both the Reuters and the 20 Newsgroups datasets. An evaluation of the selected cluster descriptors showed that our method selects informative phrases that accurately characterise the content of each cluster.

Acknowledgments

This research was partially funded by the Medical Research Council grant MR/L01078X/1 “Supporting Evidence-based Public Health Interventions using Text Mining”.

References

Ujwala Bharambe and Archana Kale. 2011. Landscape of web search results clustering algorithms. In *Advances in Computing, Communication and Control*, pages 95–107. Springer.

Claudio Carpineto, Stanislaw Osiński, Giovanni Romano, and Dawid Weiss. 2009. A survey of web clustering engines. *ACM Computing Surveys (CSUR)*, 41(3):17.

Andrew M. Dai, Christopher Olah, and Quoc V. Le. 2015. Document embedding with paragraph vectors. *CoRR*, abs/1507.07998.

Katerina Frantzi, Sophia Ananiadou, and Hideki Mima. 2000. Automatic recognition of multi-word terms: The c-value/nc-value method. *International Journal on Digital Libraries*, 3(2):115–130.

Sean Kandel, Ravi Parikh, Andreas Paepcke, Joseph M. Hellerstein, and Jeffrey Heer. 2012. Profiler: Integrated statistical analysis and visualization for data quality assessment. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 547–554.

Sun Kim, Lana Yeganova, and John W. Wilbur. 2015. Summarizing topical contents from PubMed documents using a thematic analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 805–810. Association for Computational Linguistics.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196.

Yeha Lee, Seung-Hoon Na, and Jong-Hyeok Lee. 2008. Search result clustering using label language model. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

David D. Lewis. 1997. Reuters-21578 text categorization test collection, distribution 1.0. Available at `https://archive.ics.uci.edu/ml`.

Yanjun Li, Soon M. Chung, and John D. Holt. 2008. Text document clustering based on frequent word meaning sequences. *Data & Knowledge Engineering*, 64(1):381–404.

Chenglong Ma, Weiqun Xu, Peijia Li, and Yonghong Yan, 2015. *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, chapter Distributional Representations of Words for Short Text Classification, pages 33–38. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *Workshop at International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Andriy Mnih and Geoffrey E. Hinton. 2009. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems*, pages 1081–1088.

Tingting Mu, John Y. Goulermas, Ioannis Korkontzelos, and Sophia Ananiadou. 2016. Descriptive document clustering via discriminant learning in a co-embedded space of multilevel similarities. *Journal of the Association for Information Science and Technology*, 67(1):106–133.

C. Filipe Nassif and Eduardo Raul Hruschka. 2013. Document clustering for forensic analysis: An approach for improving computer inspection. *Information Forensics and Security, IEEE Transactions on*, 8(1):46–54.

Roberto Navigli and Giuseppe Crisafulli. 2010. Inducing word senses to improve web search result clustering. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 116–126. Association for Computational Linguistics.

Stanisław Osiński, Jerzy Stefanowski, and Dawid Weiss. 2004. Lingo: Search results clustering algorithm based on singular value decomposition. In *Intelligent Information Processing and Web Mining*, pages 359–368. Springer.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. 1986. Learning representations by back-propagating errors. *Nature*, 323:533–536.

Ugo Scaiella, Paolo Ferragina, Andrea Marino, and Massimiliano Ciaramita. 2012. Topical clustering of search results. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, pages 223–232.

Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905.

Michael Steinbach, George Karypis, and Vipin Kumar. 2000. A comparison of document clustering techniques. In *KDD Workshop on Text Mining*, pages 525–526.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

Dawid Weiss. 2006. *Descriptive Clustering as a Method for Exploring Text Collections*. Ph.D. thesis, Poznan University of Technology, Poznań, Poland.

Pengtao Xie and Eric P. Xing. 2013. Integrating document clustering and topic modeling. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*.

Chengzhi Zhang. 2009. Document clustering description based on combination strategy. In *Fourth International Conference on Innovative Computing, Information and Control (ICICIC)*, pages 1084–1088. IEEE.

Peixin Zhao and Cun-Quan Zhang. 2011. A new clustering method and its application in social networks. *Pattern Recognition Letters*, 32(15):2109–2118.

Ye-Hang Zhu, Guan-Zhong Dai, Benjamin C.M. Fung, and De-Jun Mu. 2006. Document clustering method based on frequent co-occurring words. In *Proceedings of the 20th Pacific Asia Conference on Language, Informatics, and Computation (PACLIC)*, pages 442–445.

A Supplementary Material

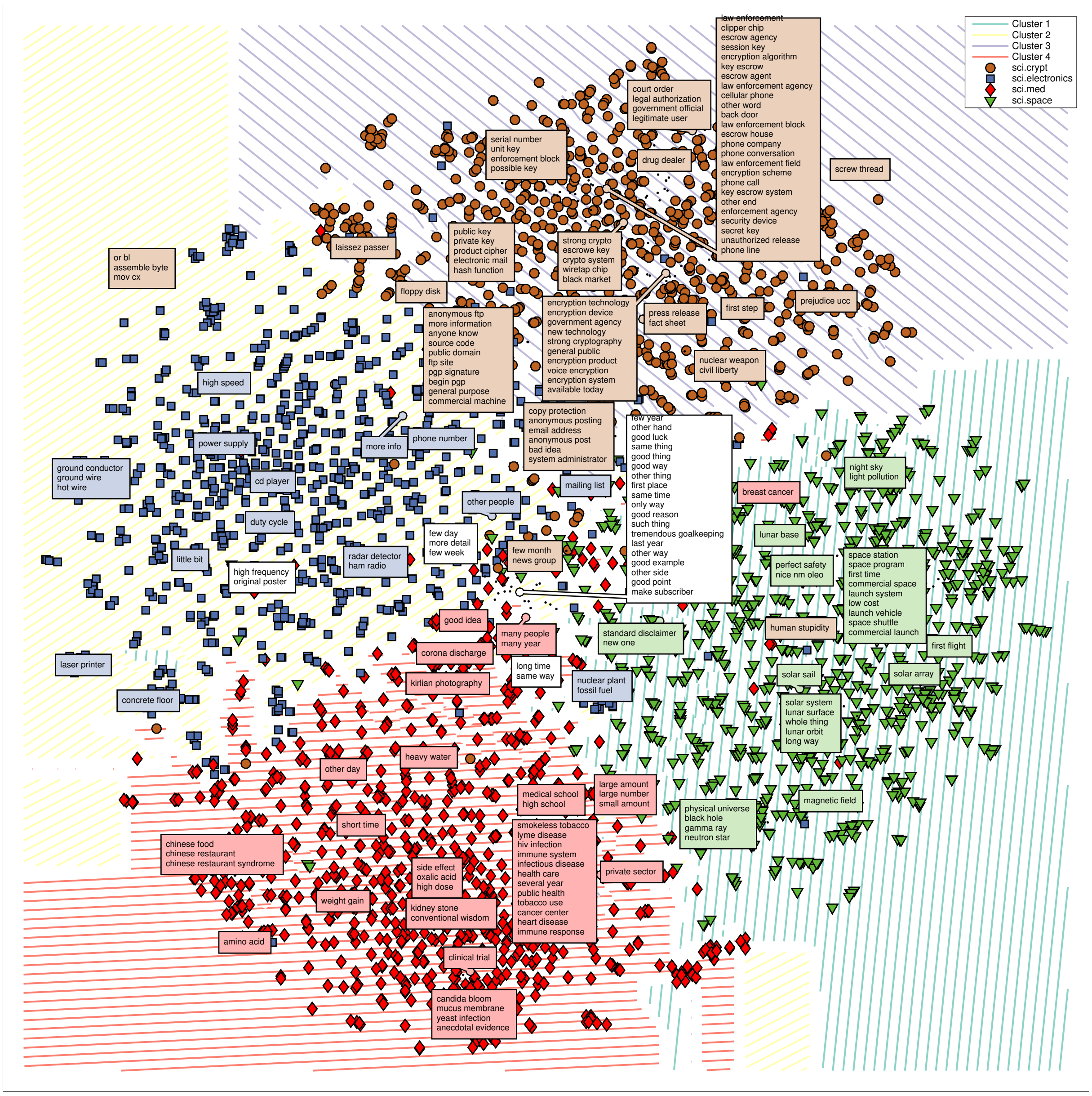

Figure 1: t-SNE visualisation of the PV-CAT co-embedding of the 20 Newsgroups ‘sci’ categories. Documents are represented by markers corresponding to the gold standard categories. Text boxes show the 200 most frequent phrases with nearby co-embedded phrases aggregated together. To show the correspondence between the categories and the phrase embedding, the text boxes are coloured based on the majority category of the documents nearest to each phrase. Sets of phrases with no majority category are left white. Hatch lines in the background denote the boundaries of each cluster, where the hatch angle (and colour) is based on the cluster. In the embedding space, each cluster is a convex set, but the t-SNE algorithm preserves local neighbourhoods and may fragment the clusters. (Best viewed with digital magnification.)