

Domain Adaptation Using a Combination of Multiple Embeddings for Sentiment Analysis

Hiroyuki Shinnou

Xinyu Zhao

Kanako Komiya

Ibaraki University, Department of Computer and Information Sciences

4-12-1 Nakanarusawa, Hitachi, Ibaraki JAPAN 316-8511

{hiroyuki.shinnou.0828, 17nm711a, kanako.komiya.nlp}
@vc.ibaraki.ac.jp

Abstract

We propose a new method for domain adaptation by using a combination of multiple embeddings for sentiment analysis. We first make the following embeddings for the document: (1) vector construction by using the bag-of-words model, (2) vector by dimension reduction using SVD, and (3) embedded vector by using doc2vec. We then connect the three embeddings. This connected vector is used as the feature vector in the learning and testing stages. In the experiment, we used an Amazon dataset that has three domains (“books”, “DVD” and “music”) and six types of domain adaptations. The experiment showed the effectiveness of our proposed method.

1 Introduction

Supervised learning has achieved great success in many natural language processing tasks. However, in real applications, the source domain S in the learning stage may be different from the target domain T in the testing stage. In this case, the classifier learned in the source domain S is ineffective in the target domain T . For example, in sentiment analysis, the classifier learned through reviews on “books” was ineffective for judging whether a review on “DVD” is positive or negative. This problem is known as the “domain shift” problem. In recent years, research on domain adaptation has been very active to solve this problem (Søgaard, 2013).

There are various types of problems in domain adaptation. However, in the case of domain adaptation of sentiment analysis, the method that maps

data in domains S and T to the shared feature subspace W is very effective.

When we conduct learning and testing on the space W , a domain shift does not occur. Therefore, the manner in which to construct W becomes the problem. For this problem, the representative research is maximum mean discrepancies (MMD) (Borgwardt et al., 2006). In recent years, many methods using deep learning have been proposed (Patel et al., 2015).

This study focuses on the sentiment analysis task, wherein the data is the document and W is the space of embeddings for the document. Furthermore, there are several methods to embed documents in the low-dimensional space, but there is no single embedding algorithm that can be used to respond to various domain adaptations from the diversity of domain differences. Therefore, we combine the embeddings of the document in this study, specifically the combination of doc2vec (Lau and Baldwin, 2016) and singular value decomposition (SVD). The document d is converted to the embedding e and g by using doc2vec and SVD respectively. Moreover we use the vector v obtained from the bag-of-words (BOW) model of the document d . Generally, v is used as the feature vector to learn the classifier. In this study, we use three connected vectors, namely $[v : e : g]$ as feature vectors instead of v .

In the experiment, we used an Amazon dataset that has three domains (“books”, “DVD” and “music”); thus we designed six types of domain adaptations. Thereafter, the method was evaluated on the basis of the average accuracy of each domain adaptation. By comparing the use of $v, [v : e]$ or $[v : g]$

as feature vectors, the effectiveness of our proposed method is demonstrated.

2 Related Work

Domain adaptation can be divided into two types: a supervised learning type that uses labeled data in the target domain and an unsupervised learning type that does not use labeled data in the target domain. In the supervised learning type, the method of Daumé’s method (Daumé III, Hal, 2007) is used as the standard method because simplicity and effectiveness.

In this study, we deal with the unsupervised learning type. The unsupervised learning can further be divided into two types: the instance-based method and feature-based method (Pan and Yang, 2010).

The instance-based method gives a weight to the instance, and weight learning is conducted. Many methods of this type assume the covariate shift which indicates that $P_S(c|\mathbf{x}) = P_T(c|\mathbf{x})$ and $P_S(\mathbf{x}) = P_T(\mathbf{x})$. Under covariate shift, the probability density ratio r is used as the weight of the instance data \mathbf{x} in the source domain. The definition of r is as follows: $r = P_T(\mathbf{x})/P_S(\mathbf{x})$. On the basis of weight learning, we obtain $P_T(c|\mathbf{x})$ (Sugiyama and Kawanabe, 2011).

The feature-based method gives a weight to the feature of the data. After giving weights, any learning method becomes available, but SVM is typically used. The problem is how to give a weight to the feature. Among feature-based methods, the most representative method is structural correspondence learning (SCL) (Blitzer et al., 2006). The feature-based method can also be used to map the data from the source domain and target domain to the feature subspace W . Among these studies, MMD is a representative method (Borgwardt et al., 2006). Furthermore, CORAL (Sun et al., 2016) is simple and effective; therefore, it has attracted considerable attention in recent years. Furthermore, the domain adaptation using deep learning is regarded a feature-based method (Glorot et al., 2011). Ref. (Sun and Saenko, 2016) shows the expanded CORAL method, and Refs. (Ganin and Lempitsky, 2015) and (Tzeng et al., 2017) show the deep learning methods for domain adaptation.

3 Domain Adaptation Using Embeddings

In this study, the target task is sentiment analysis: therefore, the data is a document. As a result, we can construct a shared feature subspace by using the embedding for the document. We construct the shared feature subspace by using doc2vec. The method constructing embedding for the document can also be seen as the dimension reduction of the document vector represented by the BOW model. Therefore, the shared feature subspace can also be constructed using the SVD.

3.1 Doc2vec

Doc2vec is a method for constructing the embedding for the document (Lau and Baldwin, 2016), and word2vec (Mikolov et al., 2013a)(Mikolov et al., 2013b) is the method for constructing the embedding of the word. Doc2vec applies the idea of word2vec to a document. Doc2vec uses two techniques, namely, Dmpv and DBoW. Dmpv is an application of CBoW in word2vec, and adds the document ID for the input vector.

On the contrary, DBoW is an application of Skip-gram in word2vec. In contrast to CBoW, Skip-gram predicts the words around the target word, and its input is a word. However, in the case of DBoW the input is not a word but a document ID.

3.2 SVD

SVD decomposes the matrix $X(M \times N)$ into three matrices as follows¹;

$$X = U\Sigma V^T.$$

If the column number of X is r , the matrix U is $M \times r$ and the column vector of X is the orthonormal basis of stretch space. Furthermore, the matrix Σ is a diagonal matrix with eigenvalues arranged in ascending order. The matrix V^T is the $r \times N$ matrix vector of matrix X , which is the orthonormal basis of stretch space. We then take X as the matrix of the document index words made from the corpus (i.e., the row vector is the document vector). If we take the matrix made from the upper k column of the matrix V as V_k , the document in the corpus can be reduced from the N dimension to the k dimension according to XV_k .

¹Here we assume $M < N$.

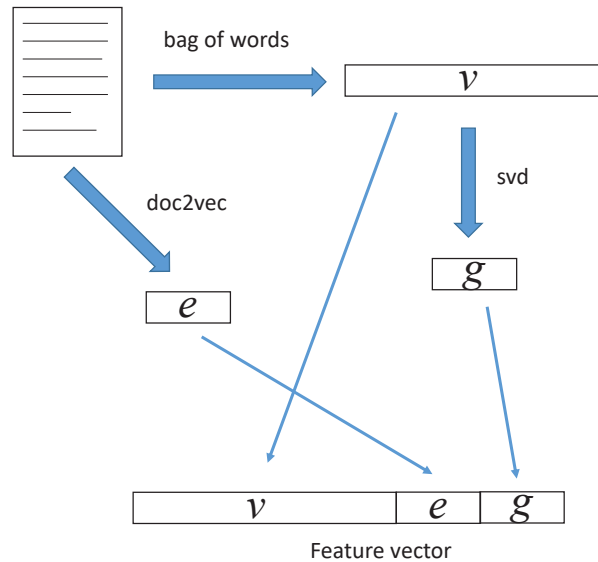


Figure 1: Feature vector connecting three embeddings

3.3 Combination of Embeddings

The embedding for the document d via doc2vec and SVD is taken as e and g respectively. In this paper, we define the vector connecting e and g as $[e : g]$. The $[e : g]$ is also an embedding of d .

The vector of d is defined as v by using the BOW model. The classifier is usually learned using the feature vector v . In this paper, we use the $[v : e : g]$ to connect the three vectors instead of v to learn the classifier (refer to Figure 1).

4 Experiments

In the experiment, we use Japanese documents in the Amazon dataset used in the study, which is available on the following site:

```
https://www.uni-weimar.de/en/media/chairs/computer-science-department/webis/data/corpus-webis-cls-10/#webis-download
```

This dataset has three domains: “books” (B), “DVD” and “music” (M). Table 1 shows the number of documents in each domain.

In the experiment, we use a method to express the document with the feature vector. By using the feature vector, an SVM classifier is learned by training

Domain	training	test	unlabeled
B (books)	2,000	2,000	169,780
D (DVD)	2,000	2,000	68,326
M (music)	2,000	2,000	55,892

Table 1: Number of document data in each domain

data in the source domain S . Thereafter, the test data in the target domain T is tested by using the classifier. The method is evaluated by the average of precisions.

The method name in Table means the the method to convert the document to the feature vector. The method expressing a document as a feature vector v by using the BOW model is referred to as “BOW”. We call the following method “D2V”; first we express the document as the 100D vector e via doc2vec. We then make the $[v : e]$ by connecting v and e . By using $[v : e]$, the classifier is learned, and we name the “SVD” as the method by using the $[v : g]$, where g is a 100D vector by using the SVD as a dimensional. Our proposed method uses the vector $[v : e : g]$ as the feature vector of the document, and we call this method “D2V+SVD.”

Table 2 and Figure 2 show the experimental results. Both D2V and SVD take advantage of embedding with a higher precision than BOW as the basic

DA	BOW	D2V	SVD	D2V+SVD
$B \rightarrow D$	0.6980	0.7245	0.7220	0.7360
$B \rightarrow M$	0.6935	0.7005	0.6885	0.7050
$D \rightarrow B$	0.6660	0.6840	0.7260	0.7155
$D \rightarrow M$	0.6910	0.6935	0.7410	0.7345
$M \rightarrow B$	0.6235	0.6370	0.6835	0.6845
$M \rightarrow D$	0.6855	0.6900	0.7205	0.7130
Average	0.6766	0.6883	0.7136	0.7148

Table 2: Result of the experiments

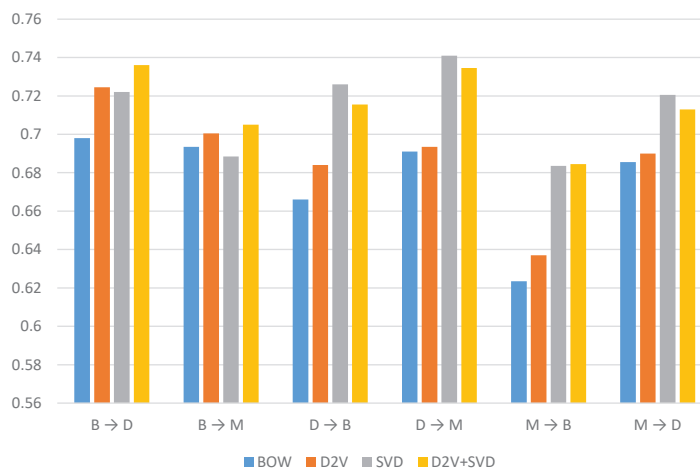


Figure 2: Result of the experiments

line. Furthermore, the proposed method is more accurate than these methods, thus indicating the effectiveness of the proposed approach.

5 Discussion

5.1 The Dimension of Embedding

In the experiments, the dimensions of embedding are set to 100 even for doc2vec and SVD. We acknowledge that there is room for improvement in doc2vec. Therefore, the above experiments were performed by changing the dimension of embeddings of doc2vec to 50 and 150. Table 3 shows the results.

In general, good accuracy is obtained at 100 dimensions, but some achieved good accuracy at 50 and 150 dimensions. The optimal dimension varies according to the type of source domain and target domain. In future, we will investigate the most suit-

able dimension.

5.2 Corpus used to Construct Embeddings

In the above experiment, we used the corpus of three corpora in three domains to construct the embeddings. Table 1 shows that the corpus in the domain “books” is significantly larger than the others. Therefore, the accuracy of $D \rightarrow B$ and $M \rightarrow B$ is improved, in which the target domain is “books” (refer to Table 4).

In this study, we discuss whether we should use all corpora or use only the source domain corpus and target domain corpus. We conduct the experiment by using only the source domain corpus and target domain corpus.

In this experiment, the embeddings are made in two corpora of the “DVD” domain and the “music” domain by using doc2vec and SVD. Table 5 shows the experimental results.

DA	BOW	D2V (50 dim.)	D2V (100 dim.)	D2V (150 dim.)
B → D	0.6980	0.7165	0.7245	0.7090
B → M	0.6935	0.6950	0.7005	0.7025
D → B	0.6660	0.6680	0.6840	0.6740
D → M	0.6910	0.7215	0.6935	0.7090
M → B	0.6235	0.6300	0.6370	0.6335
M → D	0.6855	0.6885	0.6900	0.6790
Average	0.6766	0.6866	0.6883	0.6845

Table 3: Dimension of doc2vec

DA	BOW	D2V+SVD	Rate of Improvement
B → D	0.6980	0.7360	1.054
B → M	0.6935	0.7050	1.017
D → B	0.6660	0.7155	1.074
D → M	0.6910	0.7345	1.063
M → B	0.6235	0.6845	1.098
M → D	0.6855	0.7130	1.040
Average	0.6766	0.7148	1.056

Table 4: Rate of precision improvement

The embeddings made by doc2vec, which uses the two corpora of the “DVD” domain and the “music” domain show a higher precision than the embeddings made by doc2vec, which uses corpora of “DVD”, “music” and “books” domain. On the contrary, the embeddings performed by SVD reduce the precision of the corpus of the “books” domain, i.e., for the problem of domain adaptation, only the associated corpus is used when using doc2vec and it is better to use all corpora when using SVD.

Finally, embeddings were made from the two corpora of the “DVD” domain and “music” domain by using doc2vec, and the combination of embeddings based on SVD was made using the corpora of all domains. Table 6 shows the results.

The precision of domain adaptation $M \rightarrow D$ is not changed, but the precision of domain adaptation $D \rightarrow M$ is further improved.

6 Conclusion

In this paper, we proposed a method for combining multiple embeddings for the domain adaptation of

sentiment analysis. Specifically, the document vector v was obtained from BOW model, embedding e was obtained by doc2vec, embedding g was obtained by SVD, and the vector $[v : e : g]$ connecting the three vectors is used as the feature vector. In the experiment, six types of domain adaptations are performed by using the three domains of the Amazon dataset, namely, “books”, “DVD” and “music” to show the effectiveness of our proposed method. In the future, we would like to investigate the most suitable dimension of embedding, the relationship between the embeddings, and the corpus used to construct the embeddings.

References

- John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *EMNLP-2006*, pages 120–128.
- Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. 2006. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57.
- Daumé III, Hal. 2007. Frustratingly Easy Domain Adaptation. In *ACL-2007*, pages 256–263.
- Yaroslav Ganin and Victor S. Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189.
- Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain Adaptation for Large-Scale Sentiment Classification: A Deep Learning Approach. In *ICML-11*, pages 513–520.
- Jey Han Lau and Timothy Baldwin. 2016. An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.05368*.

DA	BOW	D2V using B,D,M	SVD using B,D,M	D2V+SVD using B,D,M	D2V using D,M	SVD using D,M	D2V+SVD using D,M
D → M	0.6910	0.6935	0.7410	0.7345	0.7085	0.7155	0.7170
M → D	0.6855	0.6855	0.6900	0.7130	0.6865	0.6990	0.6960
Average	0.6882	0.6895	0.7155	0.7238	0.6975	0.7073	0.7065

Table 5: Construction of embeddings using only the relational domain corpora

DA	D2V+SVD (D2V using B,D,M SVD using B,D,M)	D2V+SVD (D2V using D,M SVD using B,D,M)
D → M	0.7345	0.7465
M → D	0.7130	0.7130
Average	0.7238	0.7298

Table 6: Construction of embeddings using only the relational domain corpus (2)

- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *ICLR Workshop paper*.
- 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.
- Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359.
- Vishal M Patel, Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. 2015. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3):53–69.
- Anders Søgaard. 2013. *Semi-Supervised Learning and Domain Adaptation in Natural Language Processing*. Morgan & Claypool.
- Masashi Sugiyama and Motoaki Kawanabe. 2011. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press.
- Baochen Sun and Kate Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops*, pages 443–450.
- Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of Frustratingly Easy Domain Adaptation. *AAAI*.
- Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. *arXiv preprint arXiv:1702.05464*.