On the Vector Representation of Utterances in Dialogue Context

Louisa Pragst, Niklas Rach, Wolfgang Minker, Stefan Ultes
Ulm University, Ulm University, Ulm University, Cambridge University
Ulm, Germany, Ulm, Germany, Ulm, Germany, Cambridge, UK
louisa.pragst@uni-ulm.de, niklas.rach@uni-ulm.de,wolfgang.minker@uni-ulm.de, su259@cam.ac.uk

Abstract

In recent years, the representation of words as vectors in a vector space, also known as word embeddings, has achieved a high degree of attention in the research community and the benefits of such a representation can be seen in the numerous applications that utilise it. In this work, we introduce dialogue vector models, a new language resource that represents dialogue utterances in vector space and captures the semantic meaning of those utterances in the dialogue context. We examine how the word vector approach can be applied to utterances in a dialogue to generate a meaningful representation of them in vector space. Utilising existing dialogue corpora and word vector models, we create dialogue vector models and show that they capture relevant semantic information by comparing them to manually annotated dialogue acts. Furthermore, we discuss potential areas of application for dialogue vector models, such as dialogue act annotation, learning of dialogue strategies, intent detection and paraphrasing.

Keywords: sentence embedding, sentence representation, dialogue embedding, dialogue representation, dialogue act detection

1. Introduction

The representation of words in vector space (Mikolov et al., 2013a), also known as word2vec, has been widely successful in the research community. It has been utilised in a number of research areas, such as machine translation (Mikolov et al., 2013b), sentiment classification (Xue et al., 2014; Zhang et al., 2015), named entity recognition (Sienčnik, 2015) and document classification (Kusner et al., 2015), among many others. Considering the success and flexibility with which word vector models can be employed in numerous applications, we aspire to investigate whether applying the word2vec approach to dialogue utterances yields similar results.

In this paper, we introduce a new language resource: dialogue vector models (DVMs), a representation of utterances in vector space that takes into account dialogue context. We investigate what adaptations are needed to the word2vec approach in order to generate such models and evaluate whether they capture information that is beneficial in a dialogue context. Furthermore, we highlight several promising areas of application that might benefit from the use of DVMs.

In the following, we discuss related work in Section 2, before introducing our approach to the generation of DVMs in Section 3. Here, we also evaluate the validity of the generated DVMs. In Section 4, we propose potential applications for those models. Finally, we draw our conclusion in Section 5.

2. Related Work

A number of different approaches to the representation of sentences in vector space have been proposed, e.g. utilising recurrent neural networks (Sutskever et al., 2014; Palangi et al., 2016), convolutional neural networks (Shen et al., 2014; Kalchbrenner et al., 2014; Hu et al., 2014) and autoencoders (Socher et al., 2011). Those approaches typically do not take into account surrounding sentences for the generation of the sentence vector, instead relying on the words in the sentence only.

Tsunoo et al. (2017) implement sentence vectors with recurrent neural networks and additionally use a bidirectional Long Short-Term Memory to capture the impact of adjacent sentences. However, this additional information is not utilised to improve the vector representation of the sentence, but to model a story transition.

Some machine translation approaches, such as (Zhang et al., 2014; Hermann and Blunsom, 2014), rely on mapping sentences in different languages into a joint vector space. Here, the correct mapping is determined taking into account not adjacent sentences, but corresponding sentences in another language.

Skip thought vectors (Kiros et al., 2015) are sentence embeddings that are generated in a similar manner as word vector representations, and therefore similar to the dialogue vector models we propose. Rather than using the words in the sentence itself as basis to create a vector representation, those vectors are generated taking into account surrounding sentences. However, this representation is trained on novels rather than dialogue. In our work, we focus specifically on dialogue and its peculiarities.

In the area of conversational response generation, neural network approaches are commonly utilised (e.g. (Sordoni et al., 2015)). Here, previous utterances in a conversation are used to generate a vector representation of the dialogue context that the response generation is based on. While the vector representation is based on adjacent sentences, a vector in such a model does not represent a singular utterance, but rather the entirety of the preceding utterances.

Cerisara et al. (2017) investigate the usability of word2vec representations for dialogue act recognition. Similarly to our work, their goal is to determine the function of an utterance in the dialogue context. In this endeavour, they use word vectors in combination with deep neural networks to determine the dialogue act of an utterance. However, the representation in vector space they utilise stays on the word level. They do not try to achieve a vector representation of the whole sentence in the dialogue context. Additionally, they target the correct assignment of dialogue labels to ut-

	SPAADIA	Switchboard
# utterances	6201	223606
# unique utterances	2903	146740
%< 5 time usage	0.97	0.99
# unique LC	2683	127539
%< 5 time usage	0.92	0.88

Table 1: Relevant data regarding the number of utterances in the employed corpora.

terances. This representation is less flexible in its potential applications than a vector representation. Lin et al. (2017) use word2vec models to implement a question answering system. However, they as well do not try to generate and exploit a vector space representation of dialogue contributions.

3. Dialogue Vector Models

A dialogue vector model is any representation of sentences as vectors that captures their semantic meaning in the dialogue context. We have performed the training and evaluation of DVMs on two dialogue corpora: the SPAADIA corpus (Leech and Weisser, 2013), which consists of task-oriented dialogues such as train travel booking, and the Switchboard corpus (Godfrey et al., 1992), which contains casual conversations on pre-specified topics.
In the following, we detail how we generate DVMs from the chosen corpora, before evaluating their capability to capture relevant semantic information. To determine the degree to which our models are generalisable, we not only evaluate their performance for one corpus, but also cross-validate the DVMs by training them on one corpus and testing them on the other.

3.1. Implementation

To generate DVMs, we adapt the generation of word representations in vector space (Mikolov et al., 2013a). The original word2vec trains its representations similarly to autoencoding. However, rather than training against the input word itself, word2vec trains words against their adjacent words in the input corpus, either using the word to predict its context or using the context to predict the word. If we consider each sentence as a single word, the same approach can easily be used to train a DVM. Therefore, existing word2vec implementations can be used and only the input needs to be modified in a manner that allows the implementation to recognise sentences as words. We employ the word2vec implementation of Deeplearning4j (Deeplearning4j Development Team, 2016) in our experiments. In the following, we describe how we modified the input text to obtain a vector representation of dialogue utterances.
The text-based approach assigns a unique identifier word to each sentence. Then, the dialogue corpus is rewritten by replacing each sentence with its identifier. Using this modified corpus as input to the original word2vec algorithm, a DVM can be trained. This approach comes with two disadvantages: first, in small corpora sentences might only occur rarely in exactly the same wording. Therefore, only little context information is available for each sentence. For reference, a word vector model, the Google News Corpus model (Mikolov et al., 2013), was trained on about 100 billion words to achieve word vectors for 3 million words. The dialogue corpora we employed in this work consist of 6201 and 223606 utterances respectively, as can be seen in Table 1. Out of those, 47/66% are unique utterances that will be assigned a dialogue vector. About 97/99% of the unique utterances are used less than five times in the dialogues, providing only little data to train the model on. The second disadvantage of this approach is its inability to generalise. Even slight alterations of a sentence, such as using synonyms or a different word sequence, leave the DVM unable to assign a dialogue vector to it if the sentence has not been encountered in this wording during the training of the model. In the case the employed dialogue corpora, this impacts the performance strongly: only 304 unique utterances from the SPAADIA corpus can be found in the Switchboard corpus, and vice versa only 109 utterances are present.
Both disadvantages of the text-based approach can be addressed by preprocessing the dialogue utterances, namely transforming the sentences into a word-based vector representation. This preprocessing applies a light clustering that groups sentences sharing the same words, thereby increasing their occurrence in the corpus data. Furthermore, common mathematical distance measures, such as the euclidean distance, can be applied to vectors to find a suitable representation for sentences that have not been encountered during training.
To implement the preprocessing, we utilise a pre-trained word vector model, the Google News Corpus model (Mikolov et al., 2013) . We obtain the word vector v_i of each word i in a sentence S and represent S as LC_S: the linear combination of its word vectors, as can be seen in Equation 1.

$$LC_S = \sum_{i \in S} v_i \quad . \tag{1}$$

A unique identifier is assigned for each linear combination (LC) and the input corpus is rewritten accordingly.
Using this representation, the percentage of utterances used less than five times can be reduced to 88/92%. Furthermore, a full mapping of utterances from one corpus to the other is achieved by replacing the a previously unencountered LC with an LC that was encountered during training and has the minimal distance from the unknown one.

3.2. Setup of the Evaluation

Projecting utterances into a vector space can only be beneficial for research in dialogue systems if it adequately captures the semantic interrelations between utterances. To ascertain that the DVM groups semantically similar dialogue contributions in close vicinity to each other, we compare a clustering based on the distances between utterances in the DVM to a clustering of those utterances based on manually assigned dialogue acts. As dialogue acts represent the meaning of an utterance in the context of the dialogue (Austin, 1962; Bunt, 1994), they are well suited as ground truth that the DVM should come close to. Hence, our evaluation comprises two steps:

1. Clustering a set of dialogue contributions based on their euclidean distance in the DVM using k-means

Test Corpus	DVM based on		Mean	SD
SPAADIA	SPAADIA	Text	0.91913	0.00126
		LC	0.91832	0.00121
	Switchboard	Text	0.50807	0.11462
		LC	0.90210	0.00404
Switchboard	Switchboard	Text	0.74648	0.00002
		LC	0.74620	0.00004
	SPAADIA	Text	0.20101	0.14877
		LC	0.73912	0.00009

Table 2: Mean and standard deviation of achieved accuracy values for different DVMs.

2. Determining the accuracy of the resulting clustering in representing manually assigned dialogue acts

Here, the accuracy A is calculated using the Rand index, defined as

$$A = \frac{TP + TN}{TP + FP + TN + FN} \quad , \tag{2}$$

where TP is the number of pairs in a cluster that share the same dialogue act, TN the number of pairs in different clusters that do not share a dialogue act, and FP and FN the number of wrongly assigned pairs, either for being clustered together but not sharing a dialogue act or for being in different clusters but having the same dialogue act.
We evaluate our approach on both text-based and LC-based DVMs. Furthermore, we employ two dialogue corpora, the SPAADIA corpus (Leech and Weisser, 2013) and the Switchboard corpus (Godfrey et al., 1992), both of which contain human-human conversation annotated manually with dialogue acts. We utilise those annotations as ground truth to which we can compare the results of our DVM. Both corpora are used for training DVMs as well as testing them. The cross-corpora evaluation ascertains the ability of DVMs to generalise. This setup results in eight test conditions to be evaluated.
For each test condition, we perform k-means clustering thirty times, with k being the number of dialogue labels used in the corpus. The resulting clusters are compared to the ground truth given by the manually annotated dialogue acts of the test data.

3.3. Results

Our evaluation shows that DVMs successfully capture dialogue relevant semantic information, and do so across corpora. Table 2 shows the mean and standard deviation of the achieved accuracy values. In the following, those results are discussed in more detail.
When testing is performed with the same corpus the models were trained on, all of them yield very good result, with over 0.91 accuracy for the SPAADIA corpus and over 0.74 for the Switchboard corpus. Furthermore, those result are achieved consistently, the standard deviation is extremely low in all cases. This suggests that the clusters are well separated and can be correctly determined by the clustering algorithm repeatedly.
When performing the evaluation on the same corpus that the DVM was trained with, the difference between DVMs trained on text and trained on LCs is minor. A slightly better performance can be reported for the text-based model for both corpora. This difference, although small, is statistically significant for the SPAADIA ($t(58) = -2.551, p = 0.013$) as well as the Switchboard corpus ($t(39.210) = -32.591, p = 0.000$). Considering only those results, employing LCs would seem unnecessary. However, their advantage of providing potential for generalisation becomes prominent when a DVM that was trained on one corpus is used for clustering utterances of the other corpus. Not only does the representation as vector allow every utterance, even if it was not present in training corpus, to be assigned a dialogue vector. The resulting representation also achieves accuracy values that are in close vicinity to the ones achieved when training and testing is performed on the same corpus, 0.9 for the SPAADIA corpus and 0.74 for the Switchboard corpus. The difference of about 0.02/0.01, although statistically significant (SPAADIA: $t(34.554) = 22.037, p = 0.000$, Switchboard: $t(31.586) = 446.239, p = 0.000$), is minor, especially considering that most of the utterances have not been encountered during training. In comparison, text-based DVMs only provide a dialogue vector for a small fraction of the test utterances and the accuracy values of text-based cross-corpus evaluation are statistically worse than those of other approaches (SPAADIA: $t(29.072) = -18.816, p = 0.000$, Switchboard: $t(29.000) = -19.812, p = 0.000$) by a large margin of about 0.4/0.5. Those results show that DVMs are able to generalise to a high degree if they are trained with LCs.
Considering the results of our evaluation, we believe that DVMs are a suitable representation of dialogue utterances and can capture the important dialogue information.
Our method of evaluation is related to the task of dialogue act recognition, which has been performed for the Switchboard corpus by Kalchbrenner and Blunsom (2013) and Cerisara et al. (2017), among others. Using supervised learning methods for dialogue classification, they achieve an accuracy 73.9% and 72.8% respectively. A good vector representation of dialogue utterances implicitly contains the information about dialogue acts. Therefore, the clusters constituted by DVMs achieve a comparable accuracy in grouping utterances according to the dialogue act. No supervision or even any particular training aimed at identifying dialogue acts was needed to achieve this result. Furthermore, DVMs can be applied more flexibly than pure dialogue act classifiers.

4. Potential Areas of Application

In the previous sections, we could show approaches to the generation of DVMs as well as the ability of those models to capture semantic interrelations between dialogue utterances. To complete our exploration of DVMs, we give an overview of research areas that we believe could benefit from utilising them in this section.

4.1. Dialogue Act Annotation

As our evaluation in Section 3.3. shows, DVMs excel as a resource for clustering algorithms to sort utterances with the same dialogue act into the same clusters. This implies

that bootstrapping manual dialogue act annotations by clustering utterances and manually assigning dialogue acts to clusters is a promising approach to reduce the work needed for dialogue corpus annotation.
In addition, automatic dialogue act recognition might be improved by the usage of DVMs. Comparing our results to current dialogue act classifiers (e.g. (Cerisara et al., 2017)) shows that a similar performance can be achieved by clustering based on DVMs. These findings support the potential of DVMs in this area.

4.2. Creation of Dialogue Policies

Machine learning approaches, in particular reinforcement learning, have become increasingly popular for training dialogue strategies in recent years (e.g. (Scheffler and Young, 2002; Rieser and Lemon, 2011)). Mathematical models form the basis of those approaches and require a numeric representation of their states. Our hypothesis is that using a meaningful representation such as DVMs as an input to those learning algorithms, rather than arbitrarily chosen ones, might be able to facilitate working with them.

4.3. Intent Detection and Paraphrasing

Indirect speech acts, as described by e.g. Searle (1975), are characterised by having, in addition to the first illocutionary act that is expressed directly by the utterance, a second one that is expressed only indirectly. Identifying the second illocutionary act and reacting accordingly is essential in a cooperative dialogue. Therefore, a lot of research goes into the automatic detection of user intent (e.g. (Allen and Perrault, 1980; Briggs and Scheutz, 2013)).
We believe that DVMs can facilitate those efforts. They project utterances that fulfil the same function in a dialogue in close vicinity to each other. This can be used to identify potential candidates for indirect speech acts as well as a corresponding direct utterances that reveal the secondary illocutionary act. This could then be followed by a check whether the proposed alternative utterance and its illocutionary act make sense in the current dialogue context.
For similar reasons, DVMs can also be used to generate more diverse dialogue contributions for the dialogue system itself. Utterances with the same functionality in a dialogue can be identified and used interchangeably by the system. The successful application of skip thoughts (Kiros et al., 2015), an approach to sentence embeddings similar to ours, for paraphrasing further supports this hypothesis.

5. Conclusion

In this work, we introduced the language resource dialogue vector model, a representation of dialogue utterances in vector space. They are inspired by the successful application of word representations in vector space and can be generated utilising existing word2vec implementations if the input is adjusted in a suitable manner. Existing word vector models can be used to preprocess the dialogue data and improve the ability of DVMs to generalise. In our evaluation, we could show that DVMs successfully project semantically similar utterances in close vicinity to each other. We presented multiple research areas in which DVMs could be successfully applied: dialogue act annotation, dialogue policy creation, intent detection and paraphrasing. The implementation and evaluation of DVM-based approaches in those areas remains future work.

6. Acknowledgements

This work is part of a project that has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 645012.

7. Bibliographical References

Allen, J. F. and Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial intelligence*, 15(3):143–178.

Austin, J. L. (1962). How to do things with words. clarendon. *Oxford*, 2005:619–650.

Briggs, G. M. and Scheutz, M. (2013). A hybrid architectural approach to understanding and appropriately generating indirect speech acts. In *AAAI*.

Bunt, H. (1994). Context and dialogue control. *Think Quarterly*, 3(1):19–31.

Cerisara, C., Kral, P., and Lenc, L. (2017). On the effects of using word2vec representations in neural networks for dialogue act recognition. *Computer Speech & Language*.

Deeplearning4j Development Team. (2016). Deeplearning4j: Open-source distributed deep learning for the JVM. *Apache Software Foundation License 2.0*.

Hermann, K. M. and Blunsom, P. (2014). Multilingual models for compositional distributed semantics. *arXiv preprint arXiv:1404.4641*.

Hu, B., Lu, Z., Li, H., and Chen, Q. (2014). Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, pages 2042–2050.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent convolutional neural networks for discourse compositionality. *arXiv preprint arXiv:1306.3584*.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966.

Lin, B.-S., Wang, C.-M., and Yu, C.-N. (2017). The establishment of human-computer interaction based on word2vec. In *Mechatronics and Automation (ICMA), 2017 IEEE International Conference on*, pages 1698–1703. IEEE.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., Song, X., and Ward, R. (2016). Deep sentence em-

bedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(4):694–707.

Rieser, V. and Lemon, O. (2011). *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation*. Springer Science & Business Media.

Scheffler, K. and Young, S. (2002). Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning. In *Proceedings of the second international conference on Human Language Technology Research*, pages 12–19. Morgan Kaufmann Publishers Inc.

Searle, J. R. (1975). *Indirect speech acts*. na.

Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. (2014). A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 101–110. ACM.

Sienčnik, S. K. (2015). Adapting word2vec to named entity recognition. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, number 109, pages 239–243. Linköping University Electronic Press.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the conference on empirical methods in natural language processing*, pages 151–161. Association for Computational Linguistics.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Tsunoo, E., Bell, P., and Renals, S. (2017). Hierarchical recurrent neural network for story segmentation. *Proc. Interspeech 2017*, pages 2919–2923.

Xue, B., Fu, C., and Shaobin, Z. (2014). A study on sentiment computing and classification of sina weibo with word2vec. In *Big Data (BigData Congress), 2014 IEEE International Congress on*, pages 358–363. IEEE.

Zhang, J., Liu, S., Li, M., Zhou, M., Zong, C., et al. (2014). Bilingually-constrained phrase embeddings for machine translation. In *ACL (1)*, pages 111–121.

Zhang, D., Xu, H., Su, Z., and Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and svm perf. *Expert Systems with Applications*, 42(4):1857–1863.

8. Language Resource References

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520. IEEE.

Leech, G. and Weisser, M. (2013). The spaadia annotation scheme. *Retrieved from martinweisser. org/publications/SPAADIA_Annotation_Scheme. pdf (last accessed November 2015)*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.