Word Re-Embedding via Manifold Dimensionality Retention

Souleiman Hasan and **Edward Curry**
Lero- The Irish Software Research Centre
National University of Ireland, Galway
{souleiman.hasan, edward.curry}@lero.ie

Abstract

Word embeddings seek to recover a Euclidean metric space by mapping words into vectors, starting from words co-occurrences in a corpus. Word embeddings may underestimate the similarity between nearby words, and overestimate it between distant words in the Euclidean metric space. In this paper, we re-embed pre-trained word embeddings with a stage of manifold learning which retains dimensionality. We show that this approach is theoretically founded in the metric recovery paradigm, and empirically show that it can improve on state-of-the-art embeddings in word similarity tasks $0.5 - 5.0\%$ points depending on the original space.

1 Introduction

Concepts have been hypothesized in the cognitive psychometric literature as points in a Euclidean metric space, with empirical support from human judgement experiments (Rumelhart and Abrahamson, 1973; Sternberg and Gardner, 1983). Word embeddings, such as GloVe (Pennington et al., 2014a) and Word2Vec (Mikolov et al., 2013), harvest observed features of the latent Euclidean space such as words co-occurrence counts in a corpus and turn words into dense vectors of a few hundred dimensions. Word embeddings have proved useful in downstream NLP tasks such as Part of Speech Tagging (Collobert, 2011), Named Entity Recognition (Turian et al., 2010), and Machine Translation (Devlin et al., 2014). However, the potential of word embeddings and further improvements remain a research question.

When comparing word pairs similarities obtained from word embeddings, to word pairs similarities obtained from human judgement, it is observed that word embeddings slightly underestimate the similarity between similar words, and overestimate the similarity between distant words. For example, in the WS353 (Finkelstein et al., 2001) word similarity ground truth:

$$\begin{aligned} sim(\text{“}shore\text{”}, \text{“}woodland\text{”}) &= 3.08 \\ < sim(\text{“}physics\text{”}, \text{“}proton\text{”}) &= 8.12 \end{aligned}$$

However, the use of GloVe 42B 300d embedding with cosine similarity (see Section 4) yields the opposite order:

$$\begin{aligned} sim(\text{“}shore\text{”}, \text{“}woodland\text{”}) &= 0.36 \\ > sim(\text{“}physics\text{”}, \text{“}proton\text{”}) &= 0.33 \end{aligned}$$

Re-embedding the space using a manifold learning stage can rectify this. Manifold learning works by estimating the distance between nearby words using direct similarity assignment in a local neighbourhood, while distance between faraway words is approximated by multiple neighbourhoods based on the manifold shape. This observation forms the basis for the rest of this paper.

For instance, using Locally Linear Embedding (LLE) (Roweis and Saul, 2000) on top of GloVe, as described in this paper, can recover the right pairs order yielding:

$$\begin{aligned} sim(\text{“}shore\text{”}, \text{“}woodland\text{”}) &= 0.08 \\ < sim(\text{“}physics\text{”}, \text{“}proton\text{”}) &= 0.25 \end{aligned}$$

Hashimoto et al. (Hashimoto et al., 2016) put word embeddings under a paradigm which seeks to recover the underlying Euclidean metric semantic space. In this paradigm, word embeddings land into a space where a Euclidean metric can be used. They show that co-occurrence counts are the results of random walk sequences in the metric space, corresponding to sentences in a corpus.

Hashimoto et al. link this to manifold learning which also seeks to recover a Euclidean space

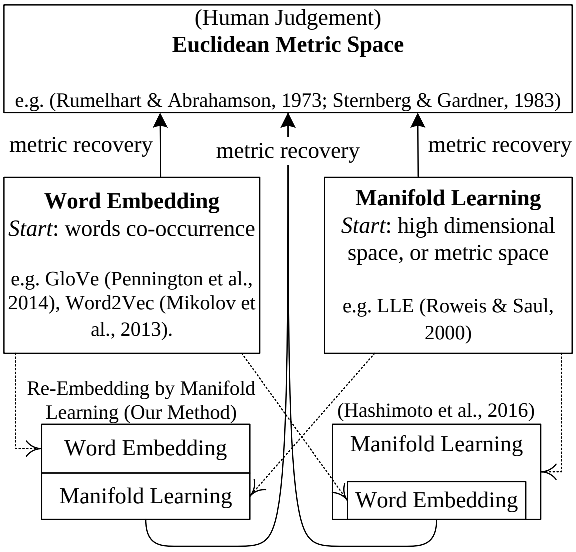

(Human Judgement)
Euclidean Metric Space
e.g. (Rumelhart & Abrahamson, 1973; Sternberg & Gardner, 1983)
metric recovery
metric recovery
metric recovery
Word Embedding
Start: words co-occurrence
e.g. GloVe (Pennington et al., 2014), Word2Vec (Mikolov et al., 2013).
Manifold Learning
Start: high dimensional space, or metric space
e.g. LLE (Roweis & Saul, 2000)
Re-Embedding by Manifold Learning (Our Method)
Word Embedding
Manifold Learning
(Hashimoto et al., 2016)
Manifold Learning
Word Embedding

Figure 1: Methodology and Related Work.

but starting from local neighbourhoods of objects, such as images or words. Global distances are built by adding up small local neighbourhoods. The authors show that word embedding algorithms can be used to solve manifold learning by generating random walks, aka sentences, on the manifold neighbourhood graph, and then embedding them.

In this work we follow a methodology which adheres to this paradigm and adopt a different angle, as per Figure 1. We start from an off-the-shelf word embedding, then we take a sample of it and feed it into manifold learning which leverages local word neighbourhoods formed in the original embedding space, learns the manifold, and embeds it into a new Euclidean space. The resulting re-embedding space is a recovery of a Euclidean metric space that is empirically better than the original word embedding when tested on word similarity tasks.

These results show that word embeddings can be improved in estimating the latent metric. Such an approach can provide new opportunities to improve our understanding of embedding methods, their properties, and limits. It also allows us to reuse and re-embed off-the-shelf pre-trained embeddings, saving time on training, while aiming at improved results in downstream NLP tasks, and other data processing tasks (Hasan and Curry, 2014; Hasan, 2017; Freitas and Curry, 2014).

Section 2 discusses the related literature to this work. Section 3 details the proposed approach. Sections 4 and 5 discuss the experiments and results. The paper concludes with Section 6.

2 Related Work

The relationship to related work is depicted in Figure 1. Word embeddings are unsupervised methods based on word co-occurrence counts which can be directly observed in a corpus. Mikolov et al. presents a neural network-based architecture which learns a word representation by learning to predict its context words (Mikolov et al., 2013). Pennington et al. proposed GloVe, which directly leverages nonzero word-word co-occurrences in a global manner (Pennington et al., 2014a).

The idea of embedding objects from a high dimensional space, e.g. images, into a smaller dimensional space constitute the area of manifold learning. For instance, Roweis and Saul present the Locally Linear Embedding (LLE) algorithm and show that pixel-based distance between images is meaningful only at a local neighbourhood scale (Roweis and Saul, 2000). Reconstructions can capture the underlying manifold of the data, and can embed the high dimensional objects, into a lower dimensional Euclidean space while preserving neighbourhoods. Other methods exist such as Isomap (Balasubramanian and Schwartz, 2002) and t-SNE (Maaten and Hinton, 2008).

Hashimoto et al. show that word embeddings and manifold learning are both methods to recover a Euclidean metric using co-occurrence counts and high dimensional features respectively (Hashimoto et al., 2016). They show that word embeddings can be used to solve manifold learning when starting from a high dimensional space. In this paper we start from a trained word embedding space, and learn a manifold from it to improve results. We do not use manifold learning to reduce dimensionality, but to transform between two equally-dimensional coordinate systems.

Other related work comes from word embedding post-processing. Labutov and Lipson use a supervised model to re-embed words for a target task (Labutov and Lipson, 2013). Lee et al. filter out abnormal dimensions from a GloVe space according to their histograms and show a slight improvement in performance (Lee et al., 2016). Mu at al. perform similar post-processing through the removal of the mean vector and vectors re-projection (Mu et al., 2017). We see manifold learning as a generic, unsupervised, nonlinear, and theoretically-founded model for post-processing that can cover linear post-processing such as PCA and normalization of vectors.

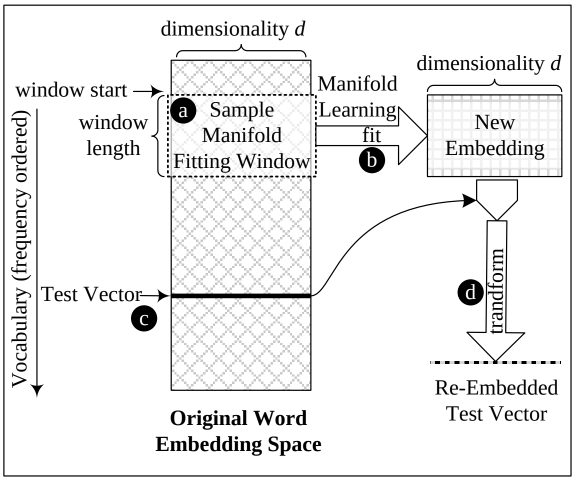


Figure 2: Re-Embedding via Manifold Learning.

3 Approach

Figure 2 illustrates our re-embedding method. We start from an original embedding space with vectors ordered by words frequencies. In step (a), we pick a sample window of vectors from this space to be used for learning the manifold. In step (b), we fit the manifold learning model to the selected sample using an algorithm such as LLE. We retain the dimensionality at this stage. In step (c), an arbitrary test vector can be selected from the original space. In step (d), the resulting fitted model serves as a transformation which can be used to transform the test vector into a vector which lives in the new re-embedding space, and used in downstream tasks.

In step (a), a sample subset of the words is used based on word frequency rank. The rational is that word embedding attempts to recover a metric space and frequent words co-occurrences can represent a better sampling of the underlying space due to their frequent usage, rather than being handled equally with other points, thus can better recover the manifold shape. Experimenting with subsets from all the vocabulary or non-frequent words, may yield no improvement. Additionally, manifold learning on all points is computationally expensive. The sampling used here follows a sliding sample window to study the effect of its start position and size. Various ways to choose a sample, e.g. random sampling, can be followed, but word frequency should remain a factor in where the sample is taken from.

In step (b), the sample is used to fit a manifold. For LLE (Saul and Roweis, 2000), that is done through learning the weights which can reconstruct each word vector from the sample X through its K-nearest neighbours in the sample, by minimizing the error function:

$$\mathcal{E}(W) = \sum_i \left| \vec{X}_i - \sum_j W_{ij} \vec{X}_j \right|^2 \qquad (1)$$

such that $W_{ij} = 0$ if $\vec{X}_j$ is not in the K-nearest neighbours of $\vec{X}_i$. The weights are then used to construct a new embedding Y of the sample X via a neighbourhood-preserving mapping through minimizing the cost function:

$$\Phi(Y) = \sum_i \left| \vec{Y}_i - \sum_j W_{ij} \vec{Y}_j \right|^2 \qquad (2)$$

In steps (c) and (d), to transform an arbitrary vector $\vec{x}$, the weights are first constructed from only the K-nearest neighbours of $\vec{x}$ in the sample X, by minimizing the function:

$$\mathcal{E}(W^x) = \left| \vec{x} - \sum_j W_j^x \vec{X}_j \right|^2 \qquad (3)$$

such that $W_j^x = 0$ if $\vec{X}_j$ is not in the K-nearest neighbours of $\vec{x}$. The weights are then used along with the new embedding Y to transform $\vec{x}$ into $\vec{y}$ which lives in the new embedding space through the equation:

$$\vec{y} = \sum_j W_j^x \vec{Y}_j \qquad (4)$$

where $\vec{Y}_j$ is the transform, from step (b), of $\vec{X}_j$ that is in the K-nearest neighbours of $\vec{x}$.

4 Experiments

Original Embedding Spaces. The original word embeddings used are pre-trained GloVe models: Wikipedia 2014 + Gigaword 5 (6B tokens, 400K vocab, 50d, 100d, 200d, & 300d vectors), and Common Crawl (42B tokens, 1.9M vocab, 300d vectors) (Pennington et al., 2014b). The vectors are ordered by the frequency of their corresponding words, so the vector representing the word 'the' comes first in the space.

Task. We use similarity tasks WS353 (Finkelstein et al., 2001) and RG65 (Rubenstein and Goodenough, 1965).

Space	Task	GloVe	Re-Embedding
6B 50d	WS353	**61.2**	56.6
6B 50d	RG65	**60.2**	53.0
6B 100d	WS353	**64.5**	64.3
6B 100d	RG65	65.3	**67.3**
6B 200d	WS353	68.5	**69.7**
6B 200d	RG65	75.5	**76.0**
6B 300d	WS353	65.8	**70.3**
6B 300d	RG65	75.5	**80.5**
42B 300d	WS353	75.2	**78.4**
42B 300d	RG65	80.0	**83.4**

Table 1: Average performance on similarity tasks. (Window start $\in [5000, 15000]$, Number of LLE local neighbours =1000, Window length = 1001, Manifold dimensionality = Space dimensionality.)

Baseline. We use the performance by the original word embeddings on the tasks. For each original space, we normalize features using their minimum and maximum values to $[-1, +1]$, and then normalize vectors to unit norms. For each pair of words in the similarity task, we get the normalized vectors and measure the cosine similarity. We finally compute the Spearman Rank Correlation with human judgements.

Approach. For a given original embedding, we normalize vectors to unit norms, then we conduct Manifold (Mfd) Re-Embedding using LLE as explained in Section 3. For each similarity task, we transform the vectors of test words into the re-embedding space before computing the cosine similarity, and the final Spearman score. We vary relevant parameters and see what effect they have on the performance, so we can understand the effectiveness of the approach and its limits.

5 Results and Discussion

Average Performance. Table 1 shows that the re-embedding method outperforms the baseline in most cases with improvements from 0.5% to 5.0%. These results are achieved for effective manifold training windows which start anywhere between 5000 and 15000. The table also shows that improvements are over spaces with underlying bigger corpora and vectors, i.e. good quality vectors which facilitate the embedding.

Manifold Dimensionality Retention. Figure 3 shows that for a given window, the re-embedding performs better when the dimensionality of the learned manifold is chosen to be closer to the original space dimensionality. In other words, dimensional reduction on the original space will bare a cost in performance.

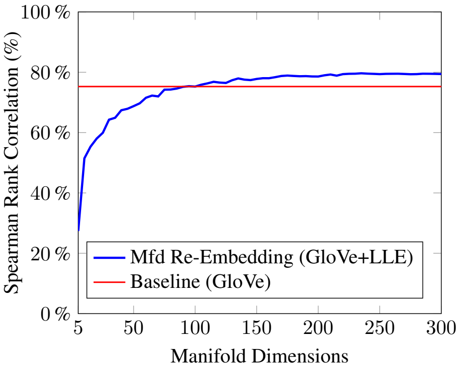


Figure 3: Accuracy on WS353 similarity task as a function of manifold dimensionality. (Space is GloVe 42B 300d. Window start = 7000, LLE local neighbours =1000, Window length = 1001.)

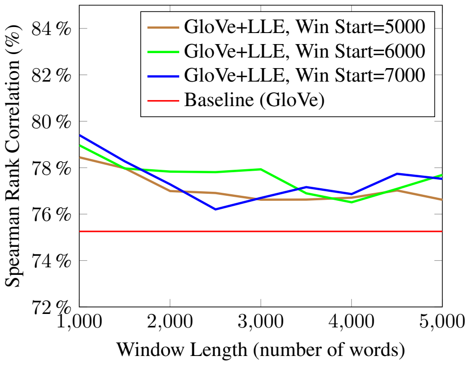
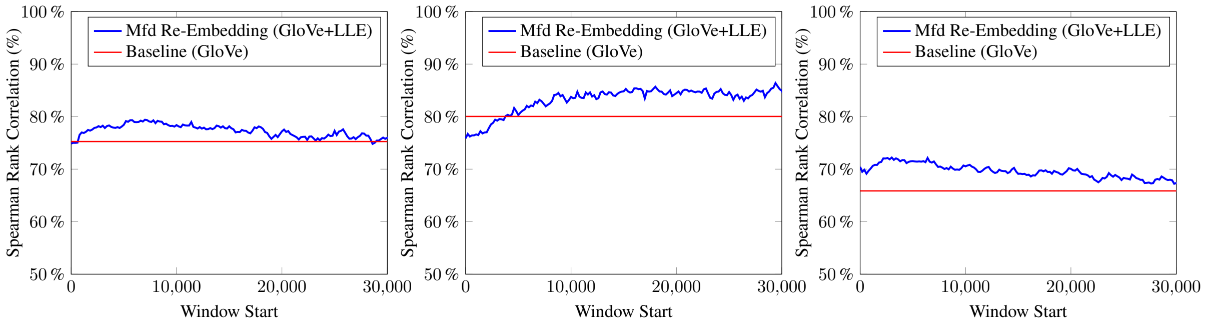


Figure 4: Accuracy on WS353 as a function of window length. (GloVe 42B 300d, LLE local neighbours =1000. Manifold dimensions =300.)

Manifold learning typically starts from a high-dimensional raw space, such as pixels, and aims to reduce the dimensionality. In our method we start from a word embedding which is already a good embedding of the raw word co-occurrences. So, dimensionality shall be retained, as suggested by Figure 3, or otherwise information can be lost during eigenvectors computation and selection in the manifold learning.

Effect of Window Length. Figure 4 shows that the best window length to choose is as close as possible to the number of local neighbours used by the manifold learning. Performance drops slightly with higher values of window length, but becomes stable after an initial drop.

Effect of Window Start. Figure 5 shows that the performance is first modest when the manifold is trained on the most frequent word vectors (i.e. stop words), but then picks up and outperforms the baseline for most cases. Performance drops grad-



(a) 42B 300d, WS353

(b) 42B 300d, RG65

(c) 6B 300d, WS353

Figure 5: Accuracy on similarity tasks as a function of window start. (a) Original space GloVe 42B 300d, with WS353. (b) 42B 300d, with RG65. (c) 6B 300d, with WS353. (LLE local neighbours =1000, Window length = 1001, Manifold dimensionality = 300.)

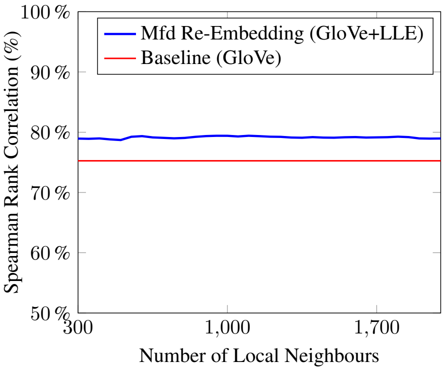


Figure 6: Accuracy on WS353 as a function of the number of manifold local neighbours. (42B 300d, Window start = 7000, Manifold dimensionality = 300, Window length = local neighbours+1.

ually as the manifold is trained on relatively less frequent word vectors.

Effect of the Number of Local Neighbours. Figure 6 shows that the performance is generally stable with variation in the number of local neighbours that the manifold is learned upon. Generally lower numbers of local neighbours mean faster manifold learning.

Discussion. The above results show that word re-embedding based on manifold learning can help the original space recover the Euclidean metric, and thus improves performance on word similarity tasks. The ability of re-embedding to achieve improved results depends on the quality of the vectors in the original space. It also depends on the choice of the window used to learn the manifold. The window start is the most influential variable, and it should be chosen just after the stop words in the original space. The choice of other parameters is relatively easier: the length of the window should be close or equal to the number of local neighbours, which in turn can be chosen from a wide range with no significant difference. The dimensionality of the original embedding space should be retained and used for learning the manifold to guarantee the best re-embedding.

6 Conclusions and Future Work

In this paper we presented a new method to re-embed words from off-the-shelf embeddings based on manifold learning. We showed that such an approach is theoretically founded in the metric recovery paradigm and can empirically improve the performance of state-of-the-art embeddings in word similarity tasks. In future work we intend to extend the experiments to include other original pre-trained embeddings, and other algorithms for manifold learning. We also intend to extend the experiments to other NLP tasks in addition to word similarity such as word analogies.

Acknowledgements

This work was supported with the financial support of the Science Foundation Ireland grant 13/RC/2094 and co-funded under the European Regional Development Fund through the Southern & Eastern Regional Operational Programme to Lero - the Irish Software Research Centre (www.lero.ie).

References

Mukund Balasubramanian and Eric L Schwartz. 2002. The isomap algorithm and topological stability. *Science*, 295(5552):7–7.

Ronan Collobert. 2011. Deep learning for efficient discriminative parsing. In *AISTATS*, volume 15, pages 224–232.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard M Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *ACL*, pages 1370–1380. Citeseer.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

André Freitas and Edward Curry. 2014. Natural language queries over heterogeneous linked data graphs: A distributional-compositional semantics approach. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pages 279–288. ACM.

Souleiman Hasan. 2017. Nosym: Non-symbolic databases for data decoupling. In *the Conference on Innovative Data Systems Research (CIDR)*.

Souleiman Hasan and Edward Curry. 2014. Thematic event processing. In *Proceedings of the 15th International Middleware Conference*, Middleware '14, pages 109–120, New York, NY, USA. ACM.

Tatsunori B Hashimoto, David Alvarez-Melis, and Tommi S Jaakkola. 2016. Word embeddings as metric recovery in semantic spaces. *Transactions of the Association for Computational Linguistics*, 4:273–286.

Igor Labutov and Hod Lipson. 2013. Re-embedding words. In *ACL*, pages 489–493.

Yang-Yin Lee, Hao Ke, Hen-Hsen Huang, and Hsin-Hsi Chen. 2016. Less is more: Filtering abnormal dimensions in glove. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 71–72. International World Wide Web Conferences Steering Committee.

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

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2017. All-but-the-top: Simple and effective postprocessing for word representations. *arXiv preprint arXiv:1702.01417*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014a. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014b. Glove resources. *Available at: http://nlp.stanford.edu/projects/glove/.*

Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326.

Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

David E Rumelhart and Adele A Abrahamson. 1973. A model for analogical reasoning. *Cognitive Psychology*, 5(1):1–28.

Lawrence K Saul and Sam T Roweis. 2000. An introduction to locally linear embedding. *Available at: www.cs.toronto.edu/%7Eroweis/lle/publications.html.*

Robert J Sternberg and Michael K Gardner. 1983. Unities in inductive reasoning. *Journal of Experimental Psychology: General*, 112(1):80.

Joseph Turian, Lev 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.