GloVeInit at SemEval-2020 Task 1: Using GloVe Vector Initialization for Unsupervised Lexical Semantic Change Detection

Vaibhav Jain
Delhi Technological University
Delhi, India
vaibhav29498@gmail.com

Abstract

This paper presents a vector initialization approach for the SemEval2020 Task 1: Unsupervised Lexical Semantic Change Detection. Given two corpora belonging to different time periods and a set of target words, this task requires us to classify whether a word gained or lost a sense over time (subtask 1) and to rank them on the basis of the changes in their word senses (subtask 2). The proposed approach is based on using Vector Initialization method to align GloVe embeddings. The idea is to consecutively train GloVe embeddings for both corpora, while using the first model to initialize the second one. This paper is based on the hypothesis that GloVe embeddings are more suited for the Vector Initialization method than SGNS embeddings. It presents an intuitive reasoning behind this hypothesis, and also talks about the impact of various factors and hyperparameters on the performance of the proposed approach. Our model ranks 12th and 10th among 33 teams in the two subtasks. The implementation has been shared publicly.[1]

1 Introduction and Background

Lexical Semantic Change (LSC) Detection is an active research topic in the field of natural language processing, and has been applied for diachronic (across time) and synchronic (across domains) tasks (Schlechtweg et al., 2019). Previously limited to manual ”close-reading” approaches, the availability of large-scale corpora have allowed the use of computational methods for this task (Tahmasebi et al., 2018). This topic has found various applications in various disciplines such as improving information retrieval from historical documents (Morsy and Karypis, 2016), preventing cross-domain ambiguity in requirements elicitation interviews (Jain et al., 2020), and studying the impact of societal and cultural changes on word meanings and usage (Tahmasebi and Risse, 2017).

LSC detection involves the use of two corpora C_1 and C_2 which, in the diachronic case, belong to different time periods t_1 and t_2 respectively. The various approaches found in the literature usually involve the construction of a word embedding space specific to each corpus. The embeddings can be constructed through count-based methods such as Positive Pointwise Mutual Information and Random Indexing, or predictive methods such as Skip-Gram with Negative Sampling (SGNS). Most of these models are stochastic in nature which means that separately trained embedding models live in their own space. In order to project them onto a *unified space*, alignment techniques such as vector initialization and orthogonal Procrustes are used. The LSC of a word is then quantitatively determined by measuring the contextual dissimilarity between the word's representations (Tahmasebi et al., 2018).

Recently, there have been efforts to evaluate these various methods by comparing their results with manually-annotated data (Schlechtweg et al., 2019; Ahmad et al., 2020). The SemEval-2020 Task 1 is one such effort which is based on LSC detection in corpora of English, German, Latin, and Swedish languages (Schlechtweg et al., 2020). Its aim is to provide an evaluation framework for unsupervised LSC detection systems by comparing their results against a ground truth, as annotated by native speakers or scholars. It consists of two subtasks: *Given two corpora C_1 and C_2 (for time periods t_1 and t_2), for a set of target words,*

[1] github.com/vaibhav29498/GloVeInit

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 208–213
Barcelona, Spain (Online), December 12, 2020.

1. *decide which words lost or gained senses between t_1 and t_2, and which ones did not; as annotated by human judges.*

2. *rank them according to their degree of LSC between t_1 and t_2 as annotated by human judges. A higher rank means stronger change.*

2 System Overview

2.1 Vector Initialization

The Vector Initialization (VI) alignment method, which was first used by Kim et al. (2014), involves the embedding space for t_1 to be trained independently on C_1. It is then used to initialize the embedding space for t_2 which is then subsequently trained from C_2. The underlying idea is that the embedding for a word w will get considerably updated if it is used within different contexts in C_1 and C_2, otherwise it will receive only a slight update. A study by Schlechtweg et al. (2019), which applied the VI method on SGNS embeddings, found it to perform significantly weaker on LSC detection tasks than other methods such as orthogonal Procrustes (OP). They attributed this to the sensitivity of the VI method to the frequency of a word in C_2. The high frequency of a word in C_2 will result into the word's embedding getting frequent updates away from its initial state. This leads to a large divergence between the word's representations even if it does not undergo any LSC from t_1 to t_2.

In a recent shared task on LSC detection in the German language, a modified VI approach achieved the third-best result and outperformed certain OP approaches (Ahmad et al., 2020). Instead of only initializing on the word embeddings obtained from C_1, the modified approach initializes on the complete SGNS model which includes the hidden layer. However, it also theoretically suffers from sensitivity to high frequency.

2.2 Comparison of GloVe and SGNS

GloVe (Global Vectors) and SGNS are unsupervised algorithms for obtaining word embedding spaces (Pennington et al., 2014; Mikolov et al., 2013). GloVe makes use of co-occurrence matrix X; its (i, j) entry, X_{ij} is the number of times the word w_j appears in the context of the word w_i (as defined by the window-size L). It trains the word embeddings by minimizing the cost function

$$J = \sum_{i=1}^{V} \sum_{j=1;j\neq i}^{V} f(X_{ij})(u_j^T v_i - \log X_{ij})^2, \tag{1}$$

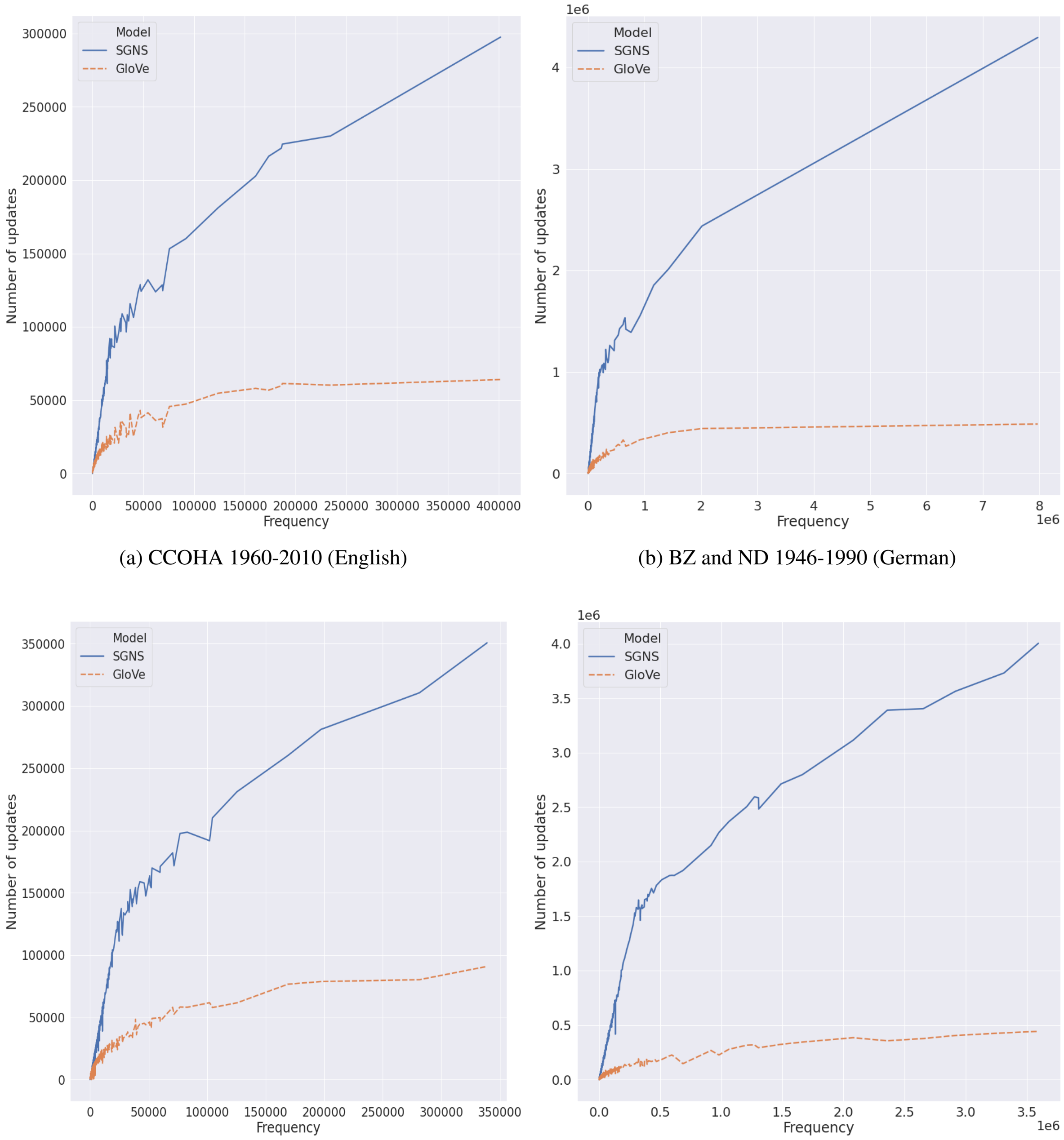
where V is the vocabulary size and $u, v \in \mathbb{R}^D$ are the word and context word vectors respectively. The final word embeddings can be obtained by summing or averaging the two.

On the other hand, SGNS is a predictive model which relies on a shallow two-layer neural network which, given a word, predicts the set of its context words. To avoid the expensive softmax function in the training objective, negative sampling is used by drawing a few negative samples from a noise distribution.

An important distinction between GloVe and SGNS from this paper's point of view is the number of updates that are made to a word embedding during training. In a single epoch, the number of updates to the SGNS embedding of a word is roughly equal to the number of words that appear in its context throughout the corpus. Hence, the number of updates is proportional to the frequency of the word which has an upper-bound of the total number of words in the corpus. However, this relationship is not exactly linear because of downsampling of frequent words and negative sampling. In the GloVe model, the number of updates received in a single epoch is equal to the number of *distinct* context words. This number is limited by an upper-bound of vocabulary size V, which is usually much less than the total number of words in the corpus.

As discussed in Section 2.1, the VI method can falsely give a high LSC score to words with high frequency. Our hypothesis is that the GloVe model is more suited for the VI method due to its lesser sensitivity to high frequency. To give an indication of the extent of this difference, the relationship between a word's frequency and the number of updates to its embedding is depicted in Figure 1 for both GloVe and

SGNS models trained on the C_2 corpora of all the four languages. These results are based on the official GloVe implementation[2] and the `gensim`[3] implementation of SGNS. The models were trained for a single epoch and only words with a frequency of greater than or equal to 5 were considered. It is clear that the proportion of updates for frequent and rare words is comparatively more balanced in the case of GloVe. On the other hand, there is a high rate of growth in the number of updates with respect to the frequency in the case of SGNS, which indicates that it is biased towards estimating a high LSC for frequent words.



(a) CCOHA 1960-2010 (English)

(b) BZ and ND 1946-1990 (German)

(c) LatinISE 0-2000 (Latin)

(d) KubHist 1895-1903 (Swedish)

Figure 1: Plot of frequency of a word (x-axis) vs the number of updates to its embedding (y-axis). The subcaptions include the name of the corpora C_2, the corresponding time-period t_2, and its language.

[2] github.com/stanfordnlp/GloVe

[3] radimrehurek.com/gensim

3 Results and Experimental Setup

The dataset provided by the competition organizers consists of the corpora C_1 and C_2, and the list of target words for four languages: English, German, Latin, and Swedish (Schlechtweg et al., 2020). All of the corpora were already prepossessed: they were in tokenized and lemmatized form with punctuation marks and one-word sentences removed.

Subtask 1 is concerned with identifying the loss or gain of one or more word sense(s), where subtask 2 tests a model's ability to detect fine-grained changes in the two sense frequency distributions. For example, consider the word *cell* which historically referred to either a chamber or the smallest unit of an organism. However, the relative usages of both these senses have decreased with time, with mobile phone being the predominant meaning now. For subtask 2, the ground-truth ranking of the target words is determined by calculating the Jensen-Shannon divergence between their normalized sense frequency distributions from t_1 and t_2 (Donoso and Sánchez, 2017). A submission is scored by its Spearman's rank-order correlation coefficient against the ground-truth ranking.

We used cosine distance as the metric to calculate the distance between a word's vectors in diachronic spaces, and every word with a distance of more than 0.55 (for German) or 0.45 (for other languages) were classified to have gained of lost word sense(s).

In all of our experiments, only words having a frequency of at least 5 were considered. Our best-performing solution during the evaluation phase achieved an accuracy of 59.9% in subtask 1 and a score of 0.352. It was obtained by training word embeddings with a dimensionality of 50 on both C_1 and C_2 for 60 epochs each with a window-size of 10. The subtask-2 score was close to but less than the modified SGNS-based VI approach discussed in Section 2.1, which was proposed by the team *IMS* and received a score of 0.372. We ranked 12th and 10th out of 33 participants in the two subtasks respectively.

Further experiments on subtask-2 were conducted after the competition for analyzing the impact of hyperparameters like window-size L and embedding dimensionality d and are reported in Table 1. Number of training epochs was limited to 20 for models with window-size 10 because of their high computational requirements. The results suggest that training the models for higher number of epochs can produce better results if the embedding dimensionality is high, but can backfire in the opposite case. Using a larger window-size improves the average result in most cases.

L	**Epochs**	***d***	**Scores**				
			English	**German**	**Latin**	**Swedish**	**Average**
5	20	5	0.012	0.366	0.391	0.15	0.23
		10	-0.144	0.362	0.394	0.114	0.182
		20	0.193	0.402	0.368	0.244	0.302
		50	**0.278**	0.446	0.315	0.229	0.317
		100	0.226	0.312	0.254	0.186	0.244
	60	5	-0.018	0.354	0.432	0.121	0.222
		10	-0.095	0.283	0.379	0.108	0.169
		20	0.212	0.41	0.377	0.324	0.331
		50	0.228	0.464	0.329	**0.366**	**0.347**
		100	0.269	0.312	0.276	0.309	0.291
10	20	5	0.011	0.334	**0.485**	0.218	0.262
		10	-0.036	0.351	0.409	0.146	0.218
		20	0.144	**0.479**	0.397	0.187	0.302
		50	0.198	0.477	0.275	0.267	0.304
		100	0.199	0.349	0.257	0.227	0.258

Table 1: Results of the post-evaluation experiments

3.1 Error Analysis

We defined the ranking error for each target word as the difference between its predicted rank (R_P) and its true rank (R_T) divided by the total number of target words (N_W). It lies in the range $(-1, 1)$ and has an ideal value of zero. A positive value indicates that the model overestimated the LSC for a word, and a negative value indicates otherwise. We then defined normalized ranking error as the observed ranking error divided by the expected absolute ranking error.

$$RankingError(R_P, R_T, N_W) = \frac{R_P - R_T}{N_W} \tag{2}$$

$$ExpectedAbsoluteRankingError(R_T, N_W) = \frac{\sum_{r_p=1}^{N_W} \frac{|r_p - R_T|}{N_W}}{N_W} \tag{3}$$

A regression plot between the relative frequency of the target words and their normalized ranking error as per our best performing model is depicted in Figure 2. A word's relative frequency is defined as its frequency divided by the total frequency of all words in the corpus.

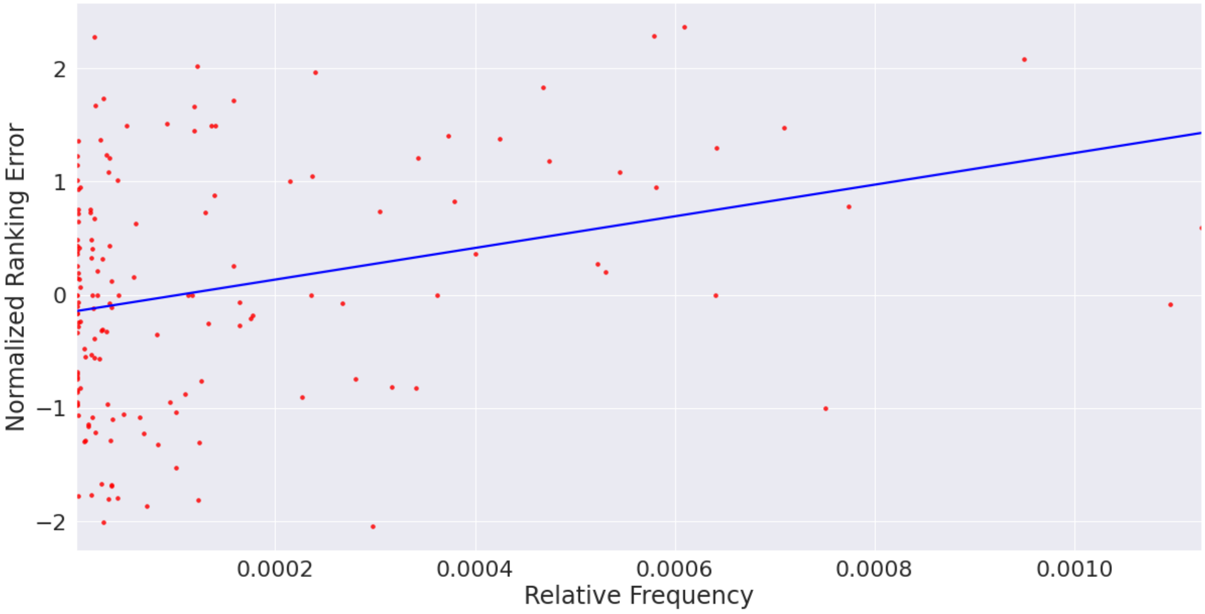


Figure 2: Regression plot between the frequency of the target words and their normalized ranking error

There is a statistically significant correlation between the normalized ranking error and the relative frequency, with the result of the Spearman's rank-order correlation test being $\rho = 0.217$ and $p = 0.007$. This indicates that the GloVe model's relative insensitivity to high frequency can lead to it assigning a rank lower than the true rank to such words. We believe that the frequency of the target words is not large enough for SGNS' sensitivity to high frequency to have a major impact on the results. This explains our model's inferior performance as compared to SGNS-based model in this task.

4 Conclusion and Future Work

In this paper, we reported our work in the SemEval2020 Task 1. We proposed a GloVe-based VI approach which achieved the 10th and 12th ranks out of 33 participating teams in the two subtasks. We gave a theoretical reasoning behind why GloVe is less sensitive towards high frequency than SGNS and thus more suited for VI method, and empirically showed the magnitude of the difference between the number of updates to word embeddings in the two models. We believe that despite a lower-than-expected performance in this competition, our work presents a good case for the suitability of GloVe model when corpora of larger size are involved. However, proving this quantitatively is a challenging task because of the limitations associated with manual annotation-based evaluation.

Planned future work includes the study of how techniques such as dimension-wise mean-centering and length normalization, which have proved beneficial in OP-based approaches, can be applied for VI method.

Acknowledgments

We would like to thank the task organizers for preparing the dataset and organizing the competition, Jens Kaiser (member of the team *IMS*) from University of Stuttgart for sharing the results from his work on the SGNS-based VI alignment method, and the anonymous reviewers for their detailed and helpful feedback.

References

Adnan Ahmad, Kiflom Desta, Fabian Lang, and Dominik Schlechtweg. 2020. Shared task: Lexical semantic change detection in german. *CoRR*, abs/2001.07786.

Gonzalo Donoso and David Sánchez. 2017. Dialectometric analysis of language variation in twitter. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 16–25, Valencia, Spain, April. Association for Computational Linguistics.

Vaibhav Jain, Ruchika Malhotra, Sanskar Jain, and Nishant Tanwar. 2020. Cross-domain ambiguity detection using linear transformation of word embedding spaces. In *Joint Proceedings of REFSQ-2020 Workshops, Doctoral Symposium, Live Studies Track, and Poster Track co-located with the 26th International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2020)*, volume 2584 of *CEUR Workshop Proceedings*, Pisa, Italy. CEUR-WS.org.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. 2014. Temporal analysis of language through neural language models. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 61–65, Baltimore, MD, USA, June. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS13, page 31113119, Red Hook, NY, USA. Curran Associates Inc.

Sara Morsy and George Karypis. 2016. Accounting for language changes over time in document similarity search. *ACM Trans. Inf. Syst.*, 35(1), September.

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, Doha, Qatar, October. Association for Computational Linguistics.

Dominik Schlechtweg, Anna Hätty, Marco Del Tredici, and Sabine Schulte im Walde. 2019. A wind of change: Detecting and evaluating lexical semantic change across times and domains. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 732–746, Florence, Italy, July. Association for Computational Linguistics.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *To appear in Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Nina Tahmasebi and Thomas Risse. 2017. On the uses of word sense change for research in the digital humanities. In *Research and Advanced Technology for Digital Libraries*, pages 246–257, Cham. Springer International Publishing.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2018. Survey of computational approaches to lexical semantic change. *CoRR*, abs/1811.06278.