Natural Language Processing Lab 1 Report: Word2Vec

Ntando Raji (2584925), Yassir Ali (2623035), and Suhail Patel (2583014)

***Abstract*—We investigate whether certain words stabilize earlier than others during the training process and examine the factors that contribute to their stabilization. By tracking a stabilization metric for each word, we analyze temporal convergence patterns and explore influences on stabilization timing, such as word frequency. Our findings reveal that word frequency alone does not explain this phenomenon, suggesting that factors like part of speech, polysemy, and syntactic roles may play a contributing role.**

I. Methods

We begin by defining the *stability* mertic. Let $E_1(w)$ and $E_2(w)$ be the k nearest neighbors of the word w in embedding spaces E_1 and E_2, then the overlap score, σ, of w is calculated as follows:

$$\sigma(w) = \frac{|E_1(w) \bigcap E_2(w)|}{k}$$

If $\sigma(w) > \epsilon$, then w is considered *stable*; otherwise, w is considered to have changed significantly between the embedding spaces [1].

We set $k = 10$ to balance capturing meaningful semantic context while avoiding noise from weakly related words. We also set $\epsilon = 0.90$ to ensure that a word is considered stable only when most of its nearest neighbors remain unchanged, while allowing minor fluctuations that naturally occur during training. Finally, the embedding at each epoch of training is compared to the final epoch, taken as the most converged state, to determine how early words reach their final positions in the space.

To evaluate word stabilization, we trained a Skip-gram Word2Vec model on the first Harry Potter book using a window size of 2. The model architecture consists of an embedding layer of size $V \times D$, where V is the vocabulary size and $D = 128$ is the embedding dimension, followed by a linear output layer projecting back to vocabulary space. Both layers were initialized with small Gaussian noise, clipped between $[-0.1, 0.1]$. Training ran for 20 epochs with batch size 256, optimizing cross-entropy loss via Adam at a learning rate of 0.001. After each epoch, embeddings were saved for analysis.

II. Results

We investigated two hypotheses on the stabilization behaviour of word embeddings in the Skip-gram Word2Vec model: (H1) some words stabilize faster than others, and (H2) high-frequency words stabilize earlier than low-frequency words. Stabilization was measured, as described iunder Methods, using nearest-neighbour overlap between embeddings at each epoch and the final epoch. .

A. H1: Distribution of Stabilization Epochs

Fig. 1 shows the distribution of stabilization epochs for all words in the vocabulary. The majority of words stabilized in later epochs, with a pronounced peak at epoch 20 where 1,710 words reached stability. However, a smaller subset of words stabilized much earlier, with some reaching stability as early as epoch 1. This confirms that there is substantial variation in stabilization speed across different words.

B. H2: Relationship Between Word Frequency and Stabilization Epoch

To examine whether word frequency influences stabilization speed, we compared stabilization epoch with log-transformed word frequency, as shown in Fig. 2. A Spearman rank correlation test yielded $\rho = 0.159$ with $p \approx 1.59 \times 10^{-34}$, indicating a very weak positive association. This suggests that while high-frequency words are sometimes assumed to stabilize earlier, frequency alone does not strongly predict the stabilization epoch in this dataset.

III. Discussion

The results for H1 (Fig. 1) confirm that there is substantial variation in stabilization speed across words. While the majority of words stabilized in later epochs, a smaller subset reached stability much earlier, with some converging as early as epoch 1. This indicates that stabilization is not uniform across the vocabulary, suggesting that properties such as the consistency of a word's contextual usage may play an important role.

For H2 (Fig. 2), the relationship between word frequency and stabilization epoch showed only a weak positive correlation ($\rho = 0.159$, $p \approx 1.59 \times 10^{-34}$). This suggests that frequency alone is not a strong predictor of stabilization speed. High-frequency words, despite being updated more often, may require more training to converge due to varied or polysemous contexts, whereas some low-frequency words stabilize earlier when their usage is contextually consistent.

Overall, these findings suggest that early stabilization is influenced more by the consistency and specificity of a word's contexts than by frequency alone. This has implications for optimizing training, as monitoring stabilization patterns could guide targeted updates for slow-to-converge words or inform early stopping criteria, reducing computational cost without sacrificing embedding quality. Due to the scope and length constraints of this report, further analysis of factors such as part-of-speech, polysemy, and syntactic roles was not included, but these remain promising avenues for future work.

IV. Figures

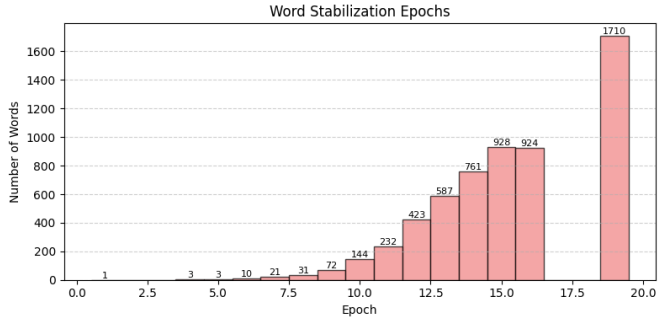

Fig. 1. Distribution of word stabilization epochs during training. The majority of words stabilize in later epochs, with a pronounced peak at epoch 19, indicating many words reach stable embeddings near the end of training.

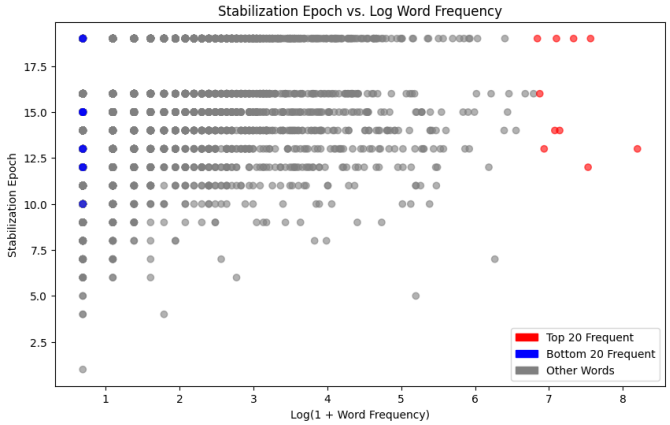

Fig. 2. Word stabilization epoch plotted against log word frequency in the Skip-gram Word2Vec model. The top 20 most frequent words (red) and bottom 20 least frequent words (blue) are highlighted to explore potential relationships between frequency and stabilization timing.

References

[1] A. Borah, M. P. Barman and A. Awekar, *Are word embedding methods stable and should we care about it?*, Proc. 32nd ACM Conf. Hypertext and Social Media, pp. 45–55, 2021.

[2] L. Wendlandt, J. K. Kummerfeld and R. Mihalcea, *Factors influencing the surprising instability of word embeddings*, arXiv preprint arXiv:1804.09692, 2018.

V. Contributions

The work presented in this lab was completed collaboratively, with each member contributing to distinct but complementary aspects of the implementation and analysis.

Ntando Raji
Designed and implemented the Skip-gram Word2Vec model in PyTorch. This involved constructing the vocabulary index mapping; that is, implementing the necessary layers to learn dense vector representations and applying the recommended initialization of weights for each layer to promote stable feature learning. Ntando implemented the training loop using mini-batch stochastic gradient descent, ensuring the model was initialised with small Gaussian-distributed weights as recommended for feature learning. He also integrated the context window generation module, allowing for flexible window size adjustments, and implemented efficient batching of input-context pairs. During training, Ntando monitored the loss trajectories of the model and validated the embedding space by querying cosine similarity between known related terms.

Yassir Ali
Led dataset preprocessing, quality assurance, and initial corpus exploration. This included reading raw text from the Harry Potter dataset, removing punctuation and non-alphabetic characters, lowercasing tokens for uniformity, and preserving the order of first occurrence in the unique word list. He implemented the skip-gram pair generation logic, ensuring correct handling of edge cases such as sentence boundaries and incomplete context windows, and verified the correctness of one-hot and index-based representations against vocabulary size and distribution. Yassir also managed initial hyperparameter exploration, varying embedding dimension, learning rate, and context window size to identify stable configurations. He worked closely with Ntando to debug training instability, adjusting batch sizes and optimiser parameters for smoother convergence.

Suhail Patel
Focused on embedding evaluation, visualisation, and interpretation of results. He implemented functions to retrieve nearest neighbours in the learned embedding space using cosine similarity, conducting targeted tests on domain-relevant terms to assess semantic clustering. He also ran controlled experiments varying the training corpus size to observe its effect on embedding quality, noting patterns of overfitting and underrepresentation for rare words. In addition to producing comparative plots, he authored the analytical section of the report, linking results to Word2Vec theory and the sub-topic explored. Suhail ensured the final report met IEEE formatting standards, carried out thorough proofreading, and coordinated the integration of all sections. All members contributed to cross-verifying results, collaborative debugging, and iterative refinement of the implementation, holding regular discussions to interpret intermediate results, align experimental design choices, and ensure the final output met the lab objectives.