The Effect of Negative Sampling on Embedding Space and Learning Speed in Skip-Gram Models

Lathitha Nongauza

I. Introduction

This paper aims to analyze the impact of k values of 2, 5, 10, 20, and 50 on the resulting embedding space and learning speed.

II. Methodology

The core of the model is a simple three-layer MLP implemented in PyTorch, designed to learn word embeddings via the Skip-Gram approach. The network consists of an input layer which receives one-hot encoded vectors representing words from the vocabulary. The input dimension equals the vocabulary size. A fully connected linear layer projects the high-dimensional sparse input into a dense embedding space of size 50. This layer's weights are initialized with small Gaussian noise. Another fully connected linear layer maps the embedding back to the vocabulary size, producing logits corresponding to scores for all vocabulary words. The output logits represent unnormalized scores used for binary classification between positive and negative word pairs during training. The text is read, split into words, lowercased, and stripped of punctuation using regular expressions. Unique words are identified and indexed to create a vocabulary. Each word is represented as a one-hot vector of length equal to the vocabulary size. For each word in the corpus (excluding the first two and last two words to allow context), a tuple is created containing the one-hot representation of this tuple structure `(center_word, leftmost_label, left_label, right_label, rightmost_label)`. This tuple structure serves as training data. For each k value, the model was trained, and resulting embeddings were clustered with k-means and visualized via PCA(Fig 1). The analysis was limited to 200 words to ensure interpretability of plots and clustering quality. The goal was to verify that semantically related words are geometrically close, ensuring embeddings capture meaningful relationships before further evaluation.

III. Results

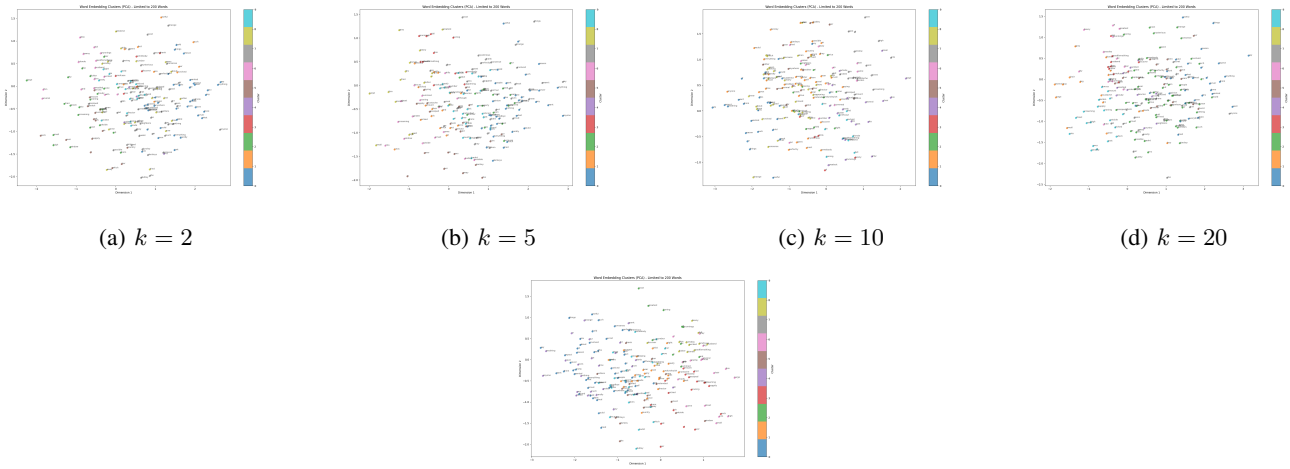
We evaluated embedding quality by computing cosine similarity between 10 pairs of related words (e.g., *she–her*, *he–him*, *mrs–mr*) and 10 pairs of unrelated words (e.g., *she–Dursleys*, *he–mysterious*). This selection was made because the corpus domain knowledge was limited (Harry Potter universe) and exhaustive evaluation was impractical. For each k and word pair set, we calculated the mean and variance of cosine similarities to holistically compare embedding quality. Related words are generally closer to each other geometrically, leading to larger dot product, conversely, dissimilar words have smaller dot products. The mean of the related and unrelated words presented are consistent with this notion(Table 1). As k increases, training time correspondingly rises due to the higher number of negative samples processed per update. Although k=50 showed the strongest separation of unrelated word pairs (lowest mean cosine similarity), the improvement over smaller values was marginal when contrasted with the substantial computational overhead incurred. Interestingly, k=2 balanced efficiency and quality well, producing the second-best distinction between unrelated words and the most stable related word similarities (lowest variance), all while achieving the fastest training..

IV. Discussion

The evaluation reveals that higher k values maintain performance on related word pairs but only slightly improve the model's ability to distinguish unrelated words, at a significantly higher training cost. Conversely, smaller k values such as 2 maintain the quality of related word embeddings and perform competitively in separating unrelated pairs, while substantially reducing training time. These findings suggest that the negative sampling hyperparameter k controls how well the model distinguishes unrelated words. Beyond a threshold (around $k = 10$), increasing k offers minimal benefit, making smaller k values more practical. For a more holistic comparison, a larger test set must be used. These insights underscore that negative sampling is a hyperparameter requiring careful tuning based on specific use cases and resource constraints. While exhaustive evaluation on larger, more diverse datasets and across various intrinsic and extrinsic benchmarks is necessary to generalize these findings, our focused analysis suggests smaller negative sample counts often strike a better balance between embedding effectiveness and training efficiency

V. Conclusion

Given the limitations of the evaluation dataset and domain knowledge, this lab concludes that using smaller negative sampling counts (e.g., $k = 2$ or 5) strikes an effective balance between embedding quality and computational efficiency. Larger k values marginally improve unrelated word separation but at the expense of slower training. In essence, k controls how well the model learns to distinguish unrelated words.



(a) $k = 2$ (b) $k = 5$ (c) $k = 10$ (d) $k = 20$

(e) $k = 50$

Fig. 1: PCA visualizations of embeddings clustered by k-means for different negative sampling counts k.

TABLE I: Cosine Similarity Statistics for Related and Unrelated Word Pairs

k	**Related (mean, variance)**	**Unrelated (mean, variance)**
2	0.4061, 0.0047	0.2709, 0.0172
5	0.4423, 0.0102	0.3093, 0.0214
10	0.3727, 0.0107	0.2914, 0.0160
20	0.4372, 0.0118	0.2941, 0.0094
50	0.3831, 0.0109	0.2137, 0.0253

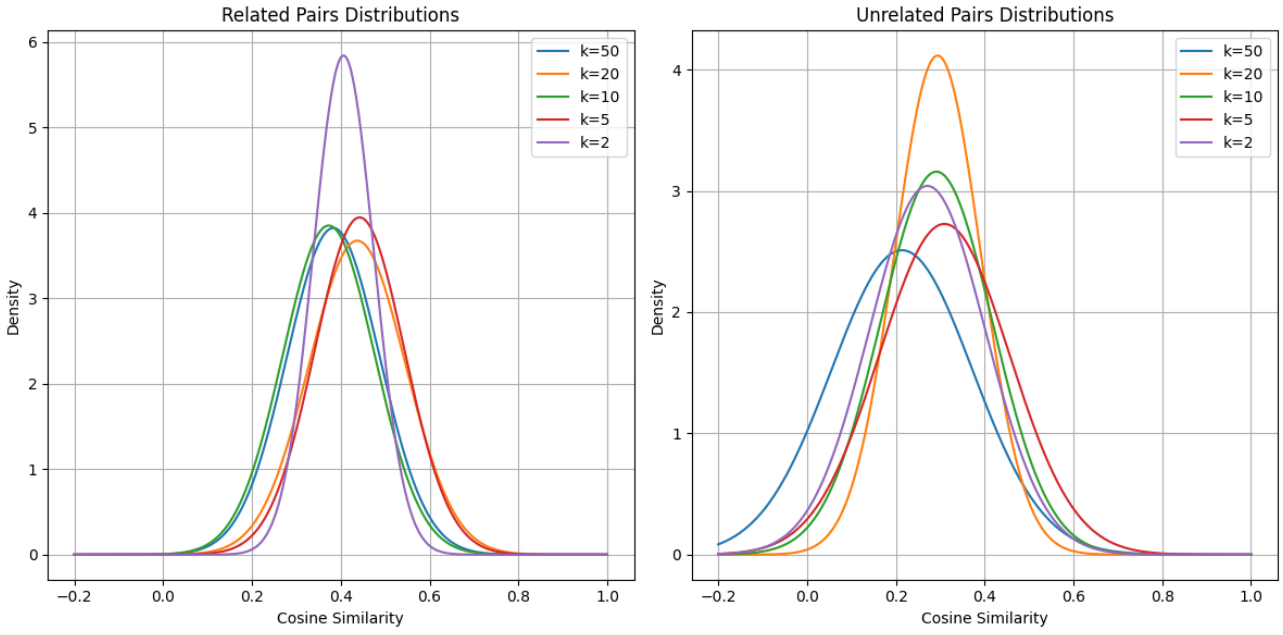


Fig. 2: Cosine similarity distributions for related and unrelated word pairs across different negative sampling counts k.