The Effect Of Using Pretrained Or None-Pretrained Tokens

Lathitha Nongauza

I. Introduction

This lab aims to analyze the impact of pre-trained or none-pretrained tokens on Harry Potter text classification.

II. Methodology

The system is structured around three main components: a Word2Vec model for learning semantic word representations, a convolutional neural network (CNN) for classifying text by book origin, and evaluation mechanisms for assessing embedding quality. The Word2Vec implementation follows the skip-gram architecture with negative sampling. The model processes the text corpus by first cleaning and sampling words from all seven books, then building a vocabulary with special handling for unknown words through an 'UNK' token. For training, it creates center-context word pairs using a window of two words on either side of each target word. The neural network consists of a simple feedforward architecture with one hidden layer that serves as the embedding layer, transforming one-hot encoded word representations into dense 50-dimensional vectors. During training, the model learns to predict context words from center words while employing negative sampling to improve efficiency and representation quality by contrasting real context words with randomly sampled negative examples. The CNN classifier leverages these learned embeddings for the book classification task. It uses an embedding layer initialized with either pre-trained Word2Vec embeddings or randomly initialized vectors, followed by a convolutional filter of size 3 to capture diverse tri-gram patterns. The model applies max-pooling over time to extract the most important features from each filter, concatenates these features, and passes them through a fully connected layer for final classification into one of the seven book categories. The training process includes dataset preparation with proper tokenization and padding, and incorporates validation-based early stopping to prevent overfitting. The CNN models were trained using an approach that monitored performance across training, validation, and test datasets. The training process employed a 70-15-15 split for training, validation, and test sets respectively.

III. Results

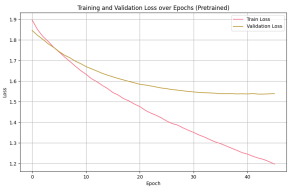
The results include the training, validation, and validation accuracy as shown in (Figure 1). We evaluated the CNN's performance by selecting two small corpora from each book and processing them through both resulting CNNs after training, as illustrated in (Figure 2). The most noticeable difference was the training speed between pre-trained and non-pretrained embeddings. The pre-trained model trained significantly faster than the non-pretrained model. We observed minimal increases in validation loss and only marginal improvements in validation accuracy throughout the training of the pre-trained model. The pre-trained model also achieved slightly higher accuracy on the test set. However, the most notable performance difference emerged in the hand-picked test set of 14 text corpora. The pre-trained model demonstrated a prediction accuracy four times greater than the non-pretrained model, as detailed in (Table 1).

IV. Discussion

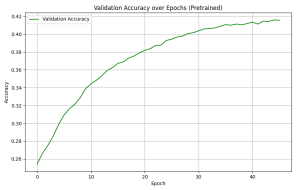
Prior to our analysis, we hypothesized that the pre-trained model would demonstrate superior performance, as its embeddings were specifically related to the classification domain. Our experimental results confirmed this hypothesis, with the pre-trained model outperforming its non-pretrained counterpart. However, we must note that the pre-trained CNN did not achieve optimal test accuracy, reaching only 57.14% (Table 1) on our hand-picked test set—significantly below our target accuracy of 80% or higher. This performance limitation can be attributed to constraints in both embedding quality and CNN architecture. Despite these limitations, both models produced results consistent with our initial hypothesis regarding the advantage of pre-trained embeddings. For more robust analysis in future work, improvements in embedding quality, CNN architecture refinement, and a more comprehensive hand-picked test set would be necessary to fully understand the comparative effects of these modelling approaches.

V. Conclusion

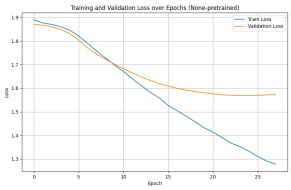
Our experimental results demonstrated that pre-trained embeddings, specifically trained on the target domain corpus, provided significant advantages in both training efficiency and classification performance. The pre-trained model converged faster during training and achieved substantially higher accuracy (four times greater) on our hand-picked test set compared to the non-pretrained model. While the pre-trained embeddings showed clear benefits, the overall classification accuracy of 57.14% indicates substantial room for improvement. These findings support the importance of domain-relevant embeddings for literary text classification tasks.



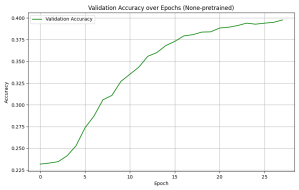
(a) Pretrained loss



(b) Pretrained accuracy

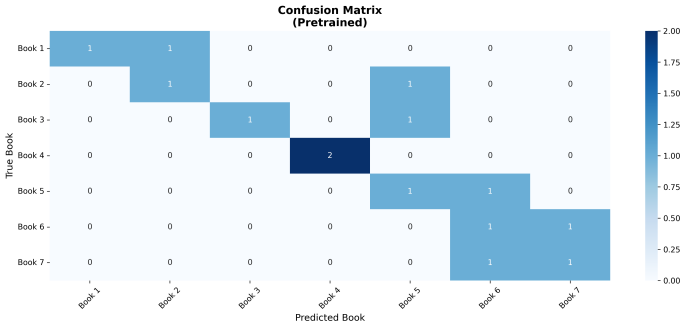


(c) None-pretrained loss

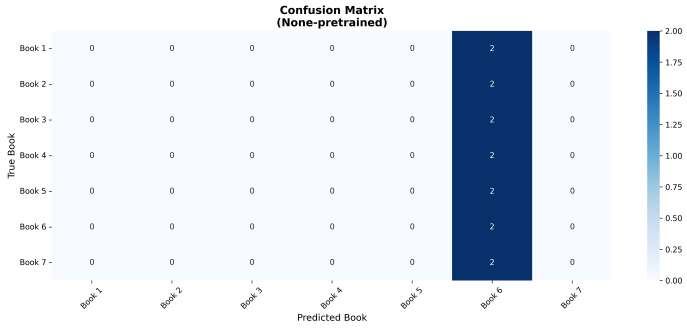


(d) None-pretrained accuracy

Fig. 1: Training and validation of pretrained and none-pretrained CNNs.



(a) Pretrained



(b) None-pretrained

Fig. 2: Confusion matrices for pretrained and none-pretrained.

TABLE I: Prediction acuracy

Accuracy	**Pretrained**	**None-pretrained**
All books	57.14%	14.29%