Summary of Alternative Approach

Overview

This document describes an alternative approach to predicting the ease of readability scores for K-12 reading samples. This algorithms approach was mostly developed by Mathis Lucka, who was the first-place winner in the Kaggle competition. Information from the Summary of Winning Algorithms document is presumed in the information below.

Additional Definitions

Cosine Similarity: Mathematically, cosine similarity is the measure of the cosine of an angle between two vectors on a multidimensional space. In practice, this measure of similarity between two vectors (i.e., excerpts from the CLEAR corpus, in our case) can be used to determine how similar two texts are irrespective of text length. A cosine value of 1 indicates two identical texts, a cosine value of 0 indicates no match between the texts, and a cosine value of -1 indicates the two texts are in the same dimension, but the meaning is opposite. For example, consider the following three texts:

I went to the Atlanta Braves baseball game yesterday.

Yesterday, I attended a baseball game at the Atlanta Braves stadium.

I did not go to the Atlanta Braves baseball game yesterday.

Humpback whales feed on shrimp-like crustaceans (krill) and small fish.

As can be seen, the first two sentences are very similar, and would have a cosine value near 1. However, the fourth sentence is much different, and if compared to either of the first two sentences, this would result in a cosine value closer to 0. Additionally, sentences one and three would have a cosine value closer to -1 because they have “opposite” meanings.

Embeddings: These are low-dimensional representations of high-dimensional vector spaces. These low-dimensional representations are used because it makes data easier to interpret and perform machine learning tasks. Embeddings can be made for individual words (i.e., word embeddings) or full sentences/texts (i.e., sentence embeddings).

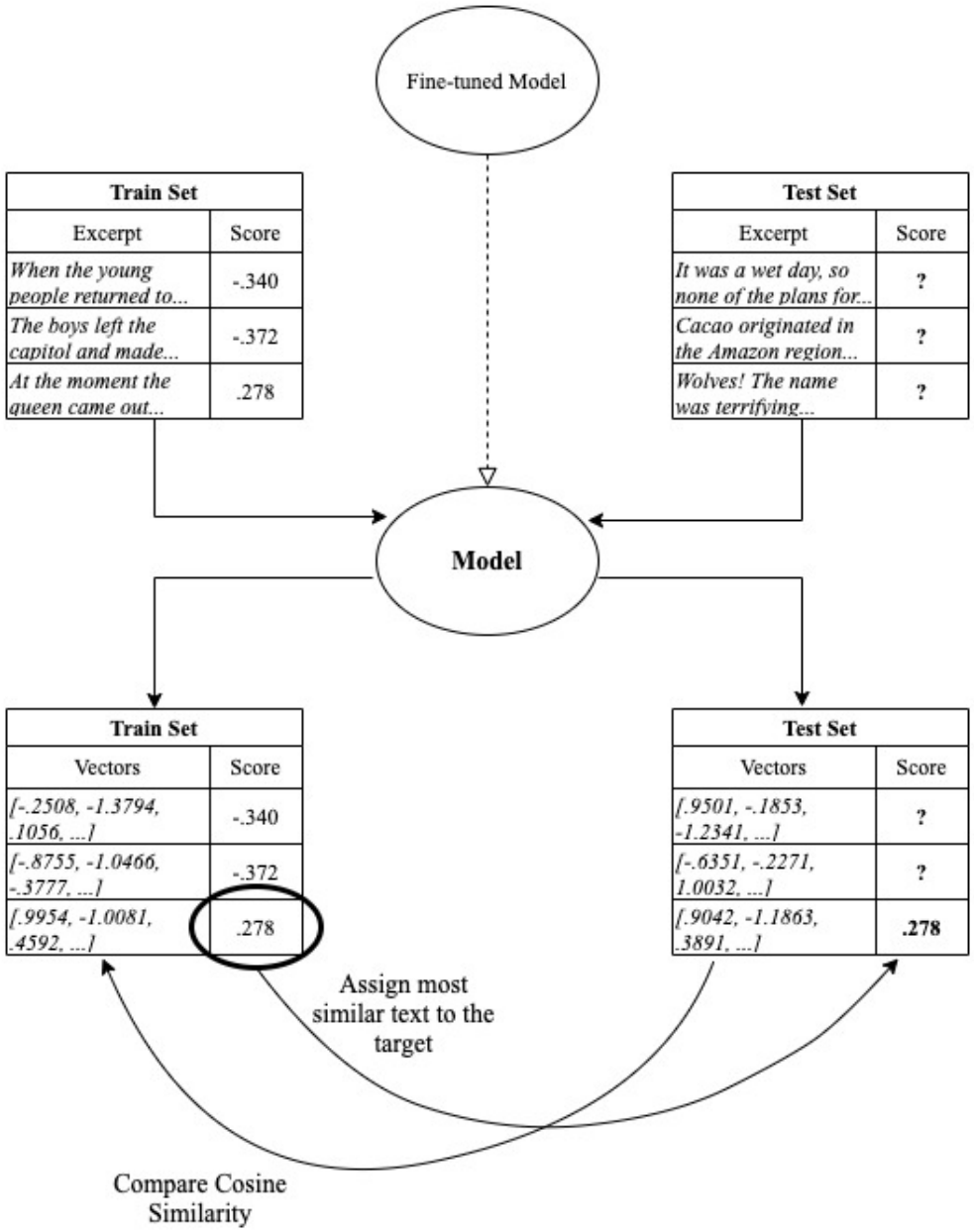
Alternative Approach

In the Kaggle competition, the winners all used an elaborate and computationally expensive transformers-based training approach to achieve the high accuracy scores. However, in a production environment, these models will likely translate into high costs, long runtime, and higher computational skills needed to edit the models.

In contrast, this alternative sentence-embeddings approach is simpler, easier to run and develop, and still achieves reasonable accuracy metrics. The basic process for this approach is as follows:

1. Calculate the sentence embeddings for each excerpt in the train and test set
2. For each excerpt in the test set, calculate the cosine similarity of each excerpt in the train set to determine which excerpt from the train set is the most similar (i.e., has a cosine similarity score closest to one).
3. Assign the known Bradley-Terry Easiness score for the most similar train excerpt to the test excerpt.
4. Repeat for each excerpt in the test set until all items have been assigned a predicted Bradley-Terry Easiness score.

A flowchart depicting this process is provided below.



Evaluation Information

This approach has the following metrics:

RMSE: .56
R-Squared: .71

A runtime test with 7,402 samples in the training set and 1,890 samples in the test set took ~3 minutes to run on a standard ML machine (i.e., Google CoLab).

Code files can be found at on my GitHub profile (yhscherber).