Multi-Genre Movie Classification from Plot Summaries with RoBERTa

University of California, Berkeley

Amy Huynh Sarah Xie

Abstract

Multi-label text classification is a complex task that has been attempted using straightforward statistical algorithms and has many applications across various fields. The following paper presents a comparison of how the powerhouse of language modeling, RoBERTa, performs against decision tree and naive Bayes classifiers, an analysis that has not yet been attempted on multi-genre movie classification from plot summaries. Findings show that RoBERTa outperforms the simpler techniques with an accuracy of 84%, macro-F1 of 53, weighted F1 of 66, and micro-F1 score of 68.

Introduction

Films, once a luxury and only able to be enjoyed from a public theater, are now ubiquitous in modern society. They tell various stories and can serve many purposes, from documentaries recorded in order to educate, fictional dramas intended for metaphoric representations of day-to-day life, to animated fantasies that spark dreams in youths. Classifying these films into informative categories, or genres, gives the audience a sense of what to expect or provides a perspective from which to view the movie from. In general, this type of document classification can be extremely informative of the subject matter (in this case, movie plot lines), as well as of the categories (genres) themselves.

However, while human beings have intuitively learned to recognize the elements of a movie's plot that align it to a specific genre, this simple task is much more difficult for a computer/machine and crucial to automating movie genre classification. People can understand context and use that context to make inferences unprompted, while a machine must be taught how to do so using data that has been specifically "cleaned" for the task and blocks out potentially "noisy" information. An efficient model that is able to predict relevant movie genres, given just a plot summary, would be extremely valuable to the industry.

The areas of study explored in this paper focus on optimizing an effective preprocessing procedure as well as finding the best model between a decision tree classifier, naive Bayes, and BERT (specifically, RoBERTa) model.

Related Work

Similar work has been done before by Blackstock and Spitz (2008) to classify movie genres using scripts scraped from the web. They leveraged a Maximum Entropy Markov Model classifier (MEMM) to model the data from the movie scripts and compared its performance to a naive Bayes algorithm on the same data, and found that the MEMM outperformed the naive Bayes algorithm. Both are very simple models and are likely unable to capture the complexity of context from language as modern-day algorithms do, and this research project hopes to improve upon that gap.

Wehrmann and Barros (2017) also have attempted a similar problem using Convolutional Neural Networks, but their training set consisted of movie trailers (image & audio multi-genre classification). Mangolin et.

al. (2006) performed a very comprehensive analysis using a multimodal dataset, composed of trailer video clips, subtitles, synopses, and movie posters. Many researchers have approached this multi-genre movie classification task with various datasets, but none, to our current knowledge, have done so using only plot summaries (text) and the BERT pre-trained language model, though Penha and Hauff (2021) have explored the conversational power of BERT in topics like movies and books. This makes our project unique.

In general, multi-label classification is an extremely useful tool and has applications in other fields aside from entertainment, including sociology and medicine. Tafreshi and Diab (2018) explore emotion detection with a multi-genre emotion corpus, and Yogarajan et. al. (2004) attempted to classify medical morbidity codes on patients from medical text. The novel techniques explored in this paper may enhance research in other fields, not just movie genre classification.

Data

This study was conducted on film data collected by Bamman, O'Connor, and Smith for their paper on the study of film characters' personas (Bamman et. al., 2013). The dataset contains metadata for 42,306 movies that were collected from scraped movie articles on Wikipedia. Although other movie features were included with this set, only the plot summary and genre columns were utilized in this project.

In the raw dataset, plot summaries ranged between 4 to 4924 words long; however, any summary that was less than 7 words long only contained links or HTML code in the description. In addition, there were a handful of movies that had no recorded genres. Both cases were promptly removed and dropped 11 records. In general, shorter summaries were either to the point and clearly implied the type of genre the film was associated with, or were vague and seemed to have little correlation with the genre. The longest summaries appeared to provide too much detail and despite explaining almost the entirety of what happens in the film, seemed to imply that the movie fit with more genres than what it is labeled as (examples provided in the Appendix).

After some early exploration on the summaries, preprocessing steps to clean the data were conducted in order to make the text usable. Cleaning steps included lowercasing, removing stopwords based on the NLTK package, and stripping away words with over 20 characters from the plots. These steps were chosen in order to present a more uniform summary to the model and remove noise in the data.

The target variable, the list of genres the movie is associated with, had very high cardinality with 363 different genres and the proportions of each genre represented is different. Table 3 in the Appendix shows the distribution of the top 10 genres. These top 10 genres are highlighted because the target label is later reduced to these 10 genres, with the addition of 1 other field, "Other", that is a placeholder label for all other genres. To further demonstrate the data available, the Appendix also contains some word cloud compilations for the top 3 genres: Drama, Comedy, and Romance. The full list of 363 genres was extremely redundant and posed a serious problem to model evaluation due its high cardinality. The imbalanced nature of the labels lead to extremely inaccurate performance metrics, so in order to have a fairer gauge on the performance, the target variables' cardinality was reduced to a more reasonable number of classes.

Methods

Three broad types of classifiers were examined in this project for their ability to correctly classify movie summaries into any and all appropriate classes according to the target feature, movie genre, provided in the dataset: Decision Tree, naive Bayes, and BERT language model. A couple of tokenization strategies (CountVectorizer and TfidfVectorizer) were tested in parallel with the classifiers as well to see which method best represented the important token features.

Decision tree classifiers are popular for their predictive power despite a relatively simplistic structure, and it was chosen to be the baseline model for the same reasons. An initial classifier was trained with the full plot

summary (after cleaning) to predict on the full set of 363 genres, but training speeds were extremely slow so a truncated 300-word plot summary version was trained after it. The 300-word model did little to speed up the process, so the corpora was shortened again to 150 words. In addition, after realizing 363 genres were not a feasible number of target variables, the labels were restructured to only the top 19 genres with an extra label to represent all the other remaining genres This classifier ran at a much more acceptable speed and delivered reasonable results.

The next classifier attempted was the naive Bayes model, also used by Blackstock & Spritz (2008) in their research as a comparison for their MEMM. naive Bayes is a commonly used classifier because of its simplicity and explainability, but has a set of strict assumptions including strong independence among the input features. While the naive Bayes models trained for this project may not meet the independence assumption and thus be overconfident, these models have repeatedly outperformed more complex models. A separate naive Bayes classifier was instantiated for each of the top 19 (and later, top 10) genres and a catch-all "other" category to perform a binary classification task predicting the presence of one of the 20 (or 11) classes. The separate predictions were combined to create an ensemble prediction determining which combination of the 20 possible genres each data sample belonged to.

The previous techniques had been popularized for many years due to their simplicity and explainability, but modern technology has opened the door to more complicated and robust models such as BERT or, more specifically for this project, RoBERTa. In their paper on RoBERTa, Liu et. al (2019) explain that they found BERT to be "significantly undertrained" and emphasize the importance of strong hyperparameters. RoBERTa builds on BERT and modifies key hyperparameters. It also removes the next-sentence pretraining objective and trains with much larger mini-batches and learning rates, thus speeding up model performance.

A dropout layer and output layer were added to the prepackaged RoBERTa model from Keras which received input IDs and attention masks from the training data tokenized with the prepackaged RoBERTa tokenizer. The model architecture and full code can be found in the Appendix.

The output layer was trained using a binary cross-entropy loss function and evaluated using binary accuracy. The predictions generated by RoBERTa are continuous between 0 and 1 for each class, representing a probability for the movie belonging to that genre. These were converted to binary targets using a threshold of 0.5 in order to be evaluated by the prepackaged accuracy and F1 metrics from by the sklearn API.

Due to limitations of machine sizes, the model could only receive a maximum of 130 words from each summary even though BERT is able to train on 512 tokens. A variety of sampling techniques were tested to optimize the 130 words extracted from each summary. First, the first 130 words were used, then 130 words were randomly sampled from each summary (without replacement), and lastly, the last 130 words were used.

While discussing the data cleaning, it was mentioned that stopwords were removed from the documents. Because of the purported importance of stopwords to grammatical structure and the flow of language, which BERT models can detect and use to inform their predictions, each of the models discussed above was trained with a version of the data that kept the stopwords included, and their predictive performance was evaluated against their counterparts with stopwords removed. All results of above methodologies are evaluated using Tensorflow's binary accuracy metric as well as macro, weighted, and micro F1 scores. They can be found in the section below.

Results

Separate training and test sets were pulled from the original dataset for model training and evaluation, respectively. The metrics used for evaluation were the binary accuracy score and three variations of the F1 score: macro, weighted, and micro. It is important to consider all 3 types of F1 score because they each weight the different classes in a multi-label classification problem differently, and offer different perspectives on model performance. Each of the RoBERTa runs were trained with just 1 epoch unless otherwise noted. The full list of experimentation can be found in Table 4 in the Appendix, but the best results from each model type (decision tree, naive Bayes, and BERT) are shown in Table 1.

Table 1: Experimentation Results for Best Models of Each Type

Model Version	Accuracy	Macro-F1	Weighted-F1	Micro-F1
CountVectorizer + Decision Tree Classification + 363 genres + truncated 150-words summary	84.2%	31.36	41.88	42.97
—				
CountVectorizer + naive Bayes + 10 (+1) genres + truncated 150-words summary	84.02%	48.93	63	64.49
—				
Final RoBERTa model (RoBERTa + reverse summary + 130 max length + 70 batch + 10 (+1) genres + 5 epochs)	84.35%	53.41	66.09	67.68

The model results show that the final version of the RoBERTa model performs the best across all 4 metrics, with an accuracy of 84%, macro-F1 of 53, weighted F1 of 66, and micro-F1 score of 68. This final version was trained using the last 130 words of each input document to predict any of the 11 target classes. Graphs of the change in accuracy and loss over each epoch can be found in the Appendix.

RoBERTa's self-attention mechanism provides token embeddings that empower the classification step to learn from the context provided by other tokens in the same document to build a more holistic understanding of the document's content. This is more powerful than the bag-of-words approach, used with the naive Bayes and decision tree models, which is just a mere count of token occurrences. This suggests that high-frequency terms are actually helpful for the genre classification task. The results above also indicate that, in general, the plot summaries contained more information at the end of the summary rather than the beginning, since the model trained using the last 130 words of each input document outperformed the models trained using the first 130 words.

It also became apparent that the models trained with only 11 target classes (10 genres and 1 "other" category) performed better than the models trained to identify the target out of 20 possible classes or even the full list (363 genres). This makes sense for a number of reasons. In the case of having 11 target classes, keeping only the 10 most frequent genres greatly increases the chances that the model makes a correct prediction because so many of the samples belong to that genre. This is especially the case for the "Drama" genre. About 45% of the samples in the training dataset fall under that category– the model could be guessing randomly, and would still make a correct prediction almost half the time. Along a similar vein, many of the genres had a lot of overlap (such as a movie that belonged to "Romance Film", "Comedy", and "Romantic Comedy", see Appendix for more examples) and the models likely had difficulty picking up on subtleties that differentiated between these categories Upon inspection of the predicted results generated by the final RoBERTa model on the test set, it becomes apparent that the model predicted "Other" for 99% of the records in the test set. More discussion on these predicted results is below.

Evaluation

The final RoBERTa model was able to correctly predict all genres for 16% of the samples in the test set, while the naive Bayes model was able to do so for only 14% of the samples. However, as shown in Table 2, the RoBERTa model tended to over-predict compared to the naive Bayes model, in the sense that it tended to associate summaries with more genres than expected.

Table 2: Proportions of Predicted Genres vs. Actual Test Data

Predicted Genre	Proportion of NB Predictions	Proportion of BERT Predictions	Proportion in Test Data
Drama	58.06%	63%	45.95%

Predicted Genre	Proportion of NB Predictions	Proportion of BERT Predictions	Proportion in Test Data
Comedy	25.29%	20.73%	24.38%
Romance Film	19.46%	13.47%	16.1%
Thriller	14.04%	22.04%	15.38%
Action	17.03%	18.91%	16.39%
World cinema	13.48%	13.06%	12.63%
Crime Fiction	6.99%	10.57%	10.13%
Horror	7.51%	13.69%	9.65%
Black-and-white	2.64%	7.2%	8.58%
Indie	1.83%	3.38%	8.54%
Other	81.8%	99.89%	79.01%

An interesting observation made during the experimentation process is that the RoBERTa model did not perform better when stopwords were included in the training data, rather the model performed better when stopwords were removed. This is counter to the existing understanding of these powerhouse language models, which are believed to extract meaning from grammar and how words are strung together in sentences, which stopwords help with. This result is likely due to the fact that the language model was unable to work at its full potential, since this project was executed in a resource-constrained environment. The modeling work was performed using Google Colab notebooks, and while a free GPU instance is provided free of charge for public Colab users, the GPU instance is shared across multitudes of users and did not allocate enough of the instance for RoBERTa to run unconstrained. This meant that 130 was the largest acceptable dimension size for the input layer before the notebook instance would crash. The dimension size could be increased if the batch size was decreased from 70, but too small of a batch size prevented the model from learning anything from the training data and resulted in low performance scores.

In light of these constraints, the results achieved by the ensemble naive Bayes models are quite favorable. While the final RoBERTa model did outperform the best naive Bayes classifier in the above experimentation, the difference in results was not large. From the perspectives of speed, simplicity of usage, and explainability the ensemble naive Bayes model developed here is a strong victor.

Conclusion

Multi-label classification is a difficult problem to solve but has salient implications for various fields beyond the entertainment industry, such as medicine and sociology. For the movie genre classification task specifically, the work done here demonstrates that in a resource-constrained environment, simple algorithms can outperform a state-of-the-art language model. With an accuracy of 84% and micro-F1 score of 68, the RoBERTa model achieved the highest performance metrics on the test set compared to all other models attempted. Aside from running the BERT model with a larger maximum length value on a GPU-enabled resource, future work should also explore techniques that would shrink the cardinality of the target class by classifying the original 363 genres in this dataset into broader genres that encompass the sub-categories, then retrain the methods discussed in this paper on the resulting dataset to see if performance improves.

Acknowledgements

We would like to thank Mark Butler, Joachim Rahmfeld, Justin Jeng, and Peter Grabowski for their guidance and support.

References

Alex Blackstock and Matt Spitz. (2008). Classifying movie scripts by genre with a MEMM using NLP-based features.

David Bamman, Brendan O'Connor, and Noah A. Smith. (2013). Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria. Association for Computational Linguistics.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv, abs/1907.11692.*.

Gustavo Penha and Claudia Hauff. (2021). What does BERT know about books, movies and music? Probing BERT for Conversational Recommendation. In *Fourteenth ACM Conference on Recommender Systems (RecSys '20), September 22–26, 2020, Virtual Event, Brazil*. ACM, New York, NY, USA, 11 pages.

Mangolin, R., Pereira, M., Britto, A. Jr., Silla, C. Jr., Feltrim, V., Bertolini, D., & Costa, Y. (2020). A multimodal approach for multi-label movie genre classification. *ArXiv, abs/2006.00654*.

Shabnam Tafreshi and Mona Diab. (2018). Emotion Detection and Classification in a Multigenre Corpus with Joint Multi-Task Deep Learning. In *Proceedings of the 27th International Conference on Computational Linguistics, pages 2905–2913, Santa Fe, New Mexico, USA*. Association for Computational Linguistics.

Jônatas Wehrmann and Rodrigo C. Barros. (2017). Movie genre classification: A multi-label approach based on convolutions through time. *Applied Soft Computing, Volume 61*, 2017, Pages 973-982, ISSN 1568-4946.

Yogarajan, V., Montiel, J., Smith, T., & Pfahringer, B. (2020). Seeing the Whole Patient: Using Multi-Label Medical Text Classification Techniques to Enhance Predictions of Medical Codes. *ArXiv, abs/2004.00430*.

Supplemental Materials

The link to the notebook containing all the code used to perform this project can be found here.

Appendix

Examples of Movie Summaries:

Example 1:

Movie: Kanteerava
Summary: The movie is a triangular love story of an orphan boy.
Genre(s): Romance Film

Example 2:

Movie: Mate Bohu Kari Nei Jaa
Summary: The film is based on two families live in Bangkok.
Genre(s): Crime Fiction, Musical, Action

Example 3:

Please note, this summary has been truncated because it was over 3,000 words long.
Movie: Genocyber
Summary: As Mel worries about Ryu, it is revealed that she is pregnant with his child. Under orders to destroy anyone opposing the mayor, the city's troops then attack the sect's church, killing everyone, including Mel, whose bullet-riddled body lies on the ground with the bodies of children surrounding her…. Grimson Rockwell: An evil mayor of the Ark de Grande who is a hypocrite, and kills anyone who opposes the law. He dies near the end when Genocyber throws his parade car into a building.
Genre(s): Animation

Word Clouds

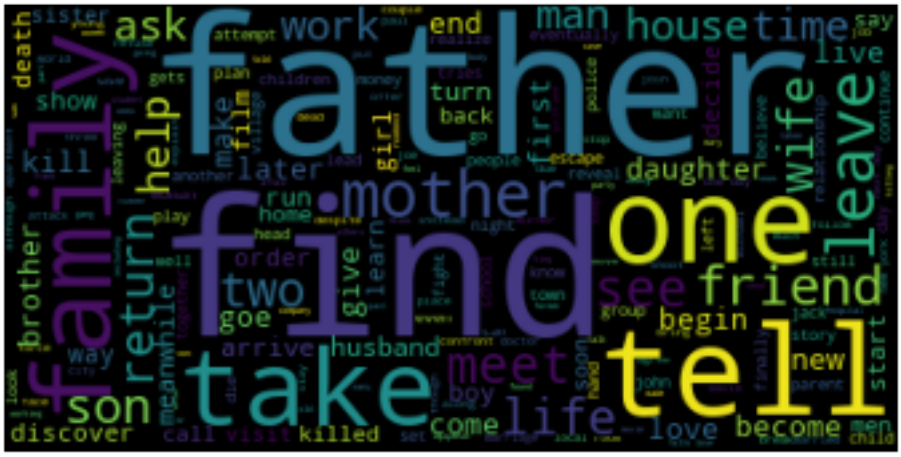

Figure 1: Word cloud of most common words found in plot summaries for Drama genres.

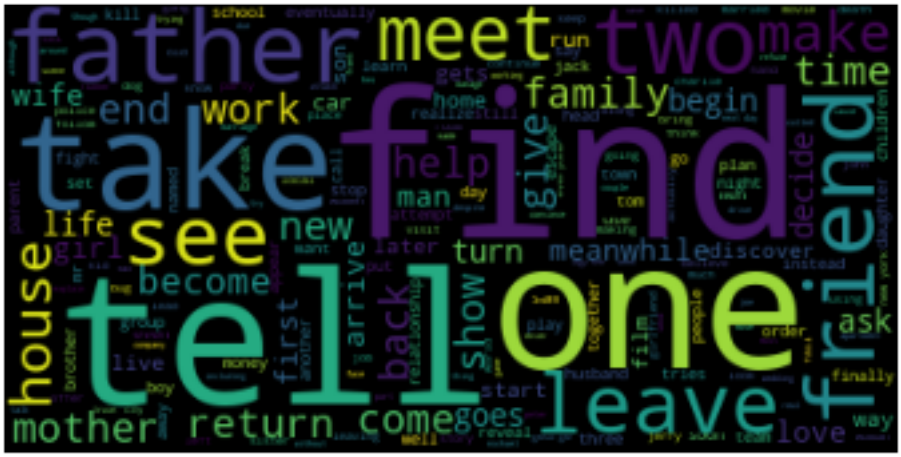

Figure 2: Word cloud of most common words found in plot summaries for Comedy genres.

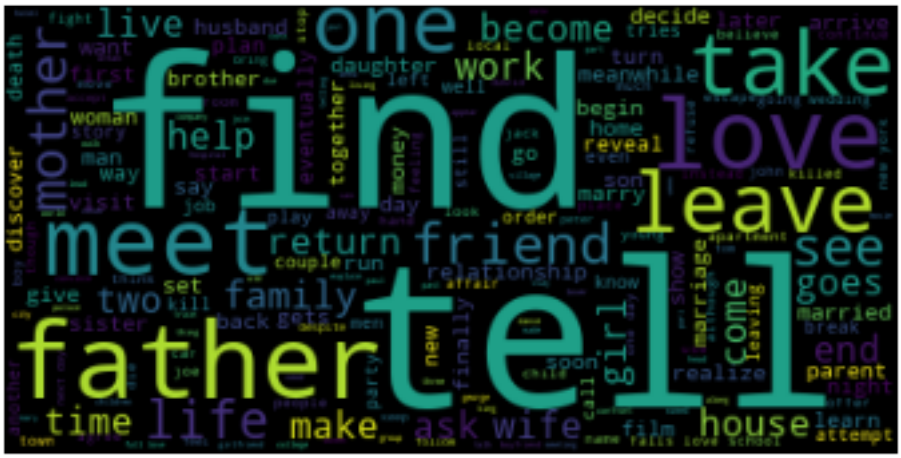

Figure 3: Word cloud of most common words found in plot summaries for Romance genres.

Distribution of Cardinality in Model Target

Table 3: Distribution of the Top 10 Genres

Genres	Count	Proportion
Other	27,148	64.33%
Drama	19,134	45.34%
Comedy	10,467	24.8%
Romance Film	6,666	15.8%
Thriller	6,530	15.47%
Action	5,868	13.9%
World cinema	5,153	12.21%
Crime Fiction	4,275	10.13%
Horror	4,082	9.67%
Black-and-white	3,731	8.84%
Indie	3,668	8.69%

Please note that any number of movies could fall into any combination of any genres, hence the sum of the proportions of all genres exceeds 100%.

Final Model Architecture

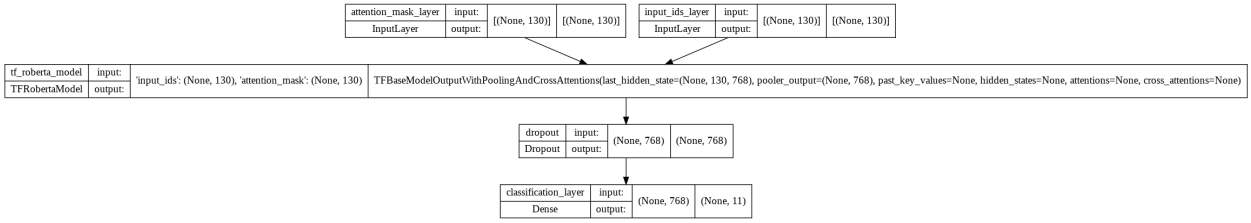

Figure 4: Roberta Model Diagram

Model Results

Table 4: Full Table of Experimentation Results

Model Version	Accuracy	Macro-F1	Weighted-F1	Micro-F1
CountVectorizer + TfidfVectorizer + Decision Tree Classification (full summary) + 363 genres	98.64%	6.97	27.49	27.99
CountVectorizer + TfidfVectorizer + Decision Tree Classification + 363 genres + truncated 300-words summary	98.63%	6.94	27.33	27.79
CountVectorizer + Decision Tree Classification (full summary) + 363 genres	98.75%	6.85	27.8	28.96
CountVectorizer + Decision Tree Classification + 363 genres + truncated 300-words summary	98.75%	6.81	27.24	28.47
CountVectorizer + Decision Tree Classification + 363 genres + truncated 150-words summary	84.2%	31.36	41.88	42.97
CountVectorizer + TfidfVectorizer + naive Bayes + 19 (+1) genres + truncated 150-words summary	88.07%	7.15	30.63	45.25
CountVectorizer + naive Bayes + 19 (+1) genres + truncated 150-words summary	88.25%	44.22	58.72	60.92
CountVectorizer + naive Bayes + 10 (+1) genres + truncated 150-words summary + truncated 150-words summary	84.02%	48.93	63	64.49
CountVectorizer + stopwords + naive Bayes + 10 (+1) genres + truncated 150-words summary	84.03%	48.7	62.88	64.43
RoBERTa + stopwords + 130 max length + 70 batch + 10 (+1) genres	82.95%	48.82	63.46	65.43
RoBERTa + 130 max length + 70 batch + 10 (+1) genres	83.18%	49.91	64.24	66.41
RoBERTa + token sampling + 130 max length + 70 batch + 10 (+1) genres	84.09%	48.96	64.31	67.26
RoBERTa + reverse summary + 130 max length + 70 batch + 10 (+1) genres	84.05%	50.6	64.57	67.16
Final RoBERTa model (previous + 5 epochs)	84.35%	53.41	66.09	67.68

Model Performance

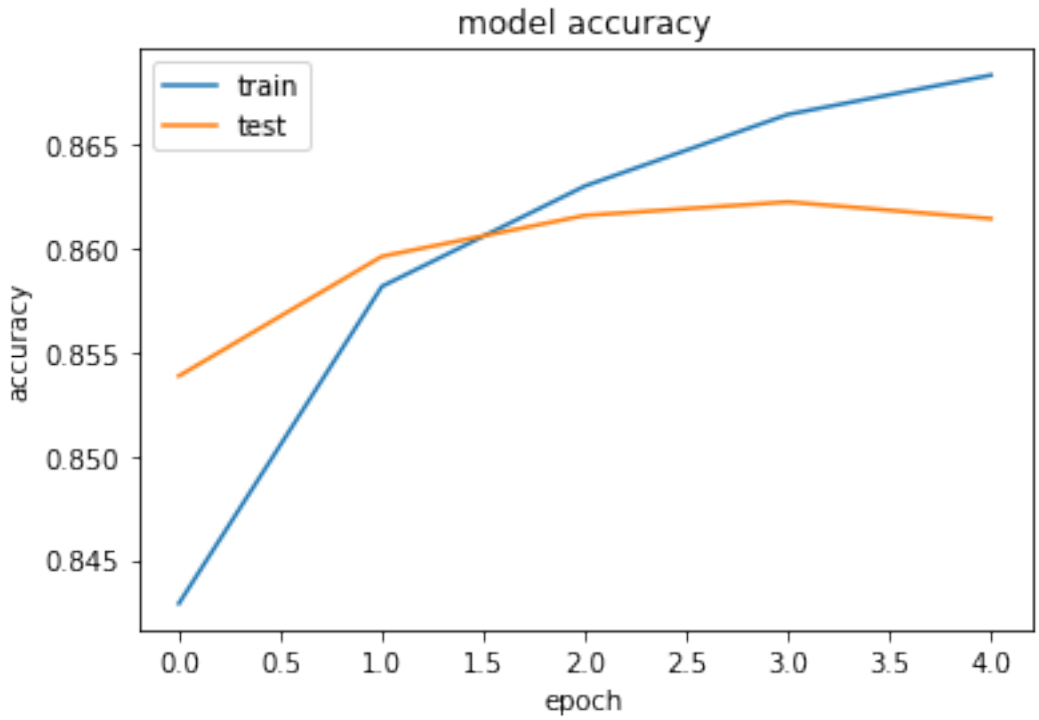

Figure 5: RoBERTa Accuracy Over 5 Epochs

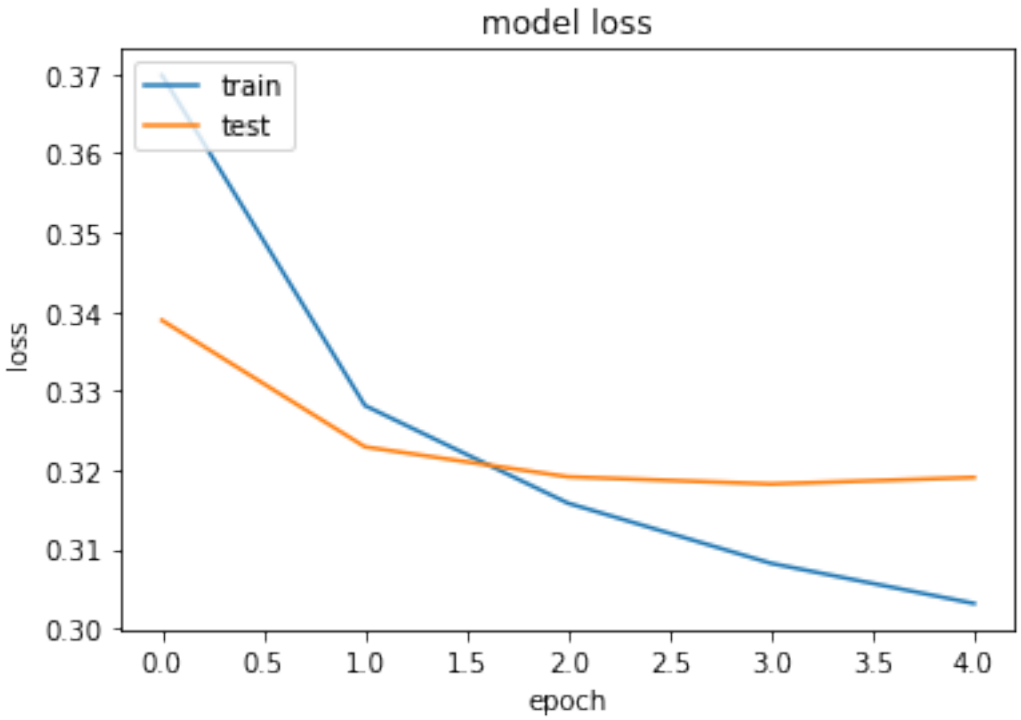

Figure 6: RoBERTa Loss Over 5 Epochs

Distribution of Similar Genres

Table 5: Distribution of Genres Similar to 'Action'

Genre	Count	Proportion
Action	5868	0.13904
Action Comedy	142	0.00336
Action Thrillers	405	0.00960
Action/Adventure	3553	0.08419

Table 6: Distribution of Genres Similar to 'Romance'

Genre	Count	Proportion
Romance Film	66666	0.15795
Romantic comedy	2075	0.04917
Romantic drama	2572	0.06094
Romantic fantasy	59	0.00140
Romantic thriller	1	0.00002