CS584: CLASSIFICATION ACROSS VARIOUS WRITTEN SUMMARY

David Fu[1]
[1]Stevens Institute of Technology
dfu6@stevens.edu

ABSTRACT

With the ever increase media content creation rate it becomes ever more difficult to classify and organize them into similar categories. Even more so as media company expand beyond single format of content. Capturing the keywords and genres type that comes from the summary/synopsis is key to be able to make categorization easier. In anticipation of applying the concept of the NLP Course the focus will be text driven classification using datasets from Kaggle with features like plot summary of movies or books. The intent is to group similar content together and provide bundle recommendation or any other use cases. Using various Natural Language Processing Model in combination with some human base rules these data will be processed and categorize into genre. The classification will help determine how similarity either a book or movies is to one another and maximize accuracy to be further able to classify other written content with summarize into respective category.

1 Introduction

The project's goal will try to train a model with genre classification abilities by utilizing the process learned in the course. By focusing on the various methods and scientific process these classification then can be use in various ways. In addition, other features outside of plot summary could be utilize to enhance the necessary feature to better optimize classification. As one is able to classified media content into a specific category no matter the type one is able to make better and more accurate recommendations across platform as one of the many use cases. For instance, it is extremely useful in mega media produce music, shows, movies, podcast, and books to be able to further introduce new content to consumers that they would enjoy. Also some of the most popular content historically speaking as comparing the modern content what genre is as long lasting as time. Also product recommendations, although it is not text based similar idea is applied using slightly different form of feature extraction such as product description or reviews.

Focusing on two distinct dataset movies and books summaries using classification techniques one will check how accurately one could group content together and the use it for utilization. Main reference will be genre along with feature extraction of keywords within summary just as zombie–undead relationship. Text classification in addition is meant to bring about structure to un-structure data. Words and sentences by themselves are often unstructured its difficult to place a sentence in a particular category without completely knowing its context. Using the summary one is better able to categorize these content.

2 Background

Much of the work involving text classification through neural language models uses various stand models such as Convolutional neural networks (CNN) for Sentence Classification or support CNN over GRU/LSTM for classification of long sentences. In addition, to achieve better performance of attention-based CNN than attention-based LSTM for answer selection. However, there are also many benchmark comparison word2vec, CNN, GRU, and LSTM in sentiment analysis and find GRU performs better.

As so many different model have been implemented by researcher for classification previously. There is no absolute best global solution model within classification. Some of the factors that have noted depends on how much the task is dependent data size, the features, and how one go about pre-process data before using it to train and validate. Also

new data and outlier how these factors gets train into the model to maintain flexibility but not hurt the accuracy. As a goal we will use these previous genre classification to complete the classifier. In addition to this, we will use this data set to make our model more robust and see if various classification for this complex data set could be as accurate. This is inspired by some neural machine translation model like Seq2seq. We will make improvement basing on the previous work.

Using classification techniques such as Bayes, K nearest neighbor, and other feature extraction one will be able to classify a new summary. Also in combination classification of the keywords extracted from the summary with the genre hoping to provide a good key indicator of content recommendation. There will also be comparison between some of the most used classification strategy to see if one is more optimal than another when it's used on different media content such the common Bayes Classifier or support vector machine. When looking at classification it may also be applied in other perspective such as image recognition to be able to find pattern and trends as it is quicker by association. Using techniques like Naive Bayes, Support Vector Machines or Neural network it will try to classify each plot summary. Along side basic technique like TFIDF various keywords within the summary will be added to the Genre list along side the existing classification.

3 Data

Using the Genres and Plot from these dataset one is able to categorize them in genres that is then verified against the existing label. Since label may differ between books and movies pre-processing will be required to apply some human rules. Also new features besides the existing Genre will be added. **Movie Summary from Wikipedia** About 35,000 movie summaries Features Provided: *Feature to be used*

Release Year / Origin/Ethnicity / Director / Actors / Wiki Page

Movie title Movie Genre Plot

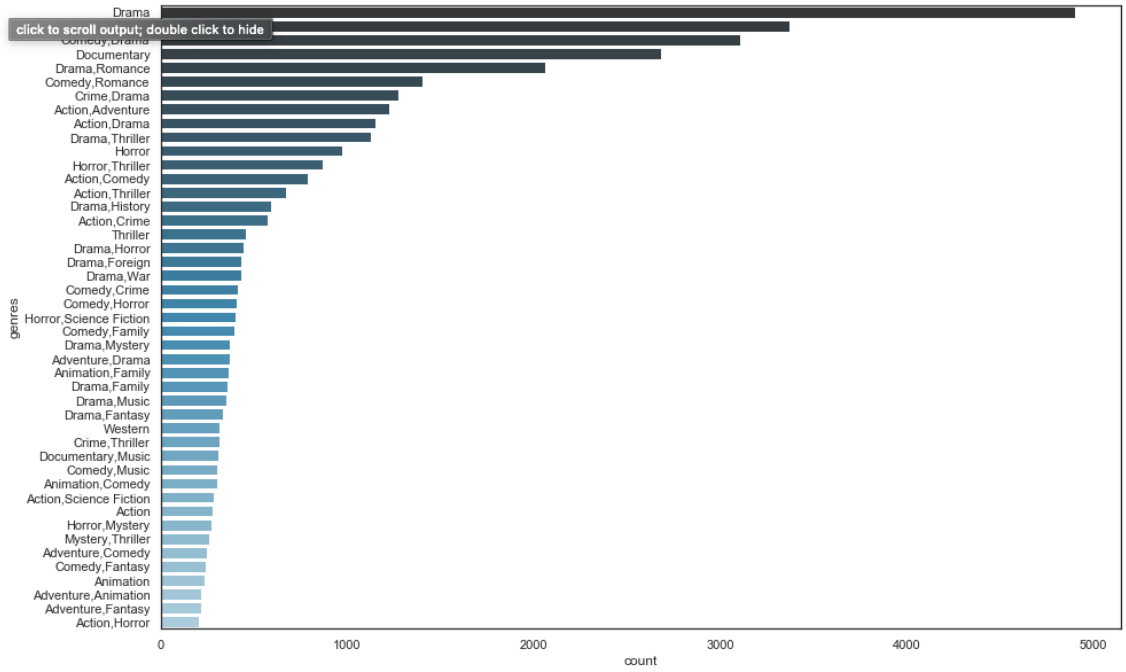
Book Summary from Wikipedia About 17,000 book summaries Feature Provided: *Feature to be used*

Wikipedia ID / Freebase ID / Book author / Publication Date /

Book Title / Genres / Plot Summary

Comparing various feature extraction method such as simple count, TF-IDF, pre-Trained TF-IDF, Word2Vec/Paragraph2Vec, or N-gram model of the plot in the data set.

3.1 Initial Data Analysis



Looking at the initial data and what each of the 40,000 movie in initially categorize into their respective data. It seems that drama is most popular genre and the best dual genre all involves drama of some form

4 Methods

In order to achieve the two part goal of classification then recommendation. First we have to extract the feature and add the keyword either by occurrence or importance to existing categorization in addition to genre. The data will then be split both randomly and intentionally into training and validation set along with a third set of external data use for prediction. Four main model will be tested for classification Naive Bayes, Linear Support Vector, Logistic Regression, or LTSM. Their accuracy compare to the base given genre will be tested without the keywords. With the best model an additional recommendation ranking of comparing most similar content already in the processed data with new data. Selective data will be used for training and prediction such as Lord of the Ring where given a book summary of the first book the ranking similarity one should be the movie equivalent follow by the sequels either in book or movie form.

5 Experimental Design

After analyzing the classification data without sufficient evidence were dropped from the training set. To focus on 20 key group of genre classification.

5.1 Logistic Regression

Logistic regression is a simple and easy to understand classification algorithm, and Logistic regression can be easily generalized to multiple classes. And using One vs many methodology one is able to train the data to be classified. Using this as baseline classification method to be then compare to future items. Logistic regression measures the relationship between the genre variable and the feature extract from the body of the plot summary by estimating probabilities using a logistic function. By the cumulative distribution function of the distribution. This method assumes there is a standard logistic distribution of errors thus have the ability to minimize the difference resulting in the category being defined

5.2 LinearSVC

Support Vector Classifier is a specific type of SVM is to fit to the data you provide, returning a ”best fit” area that divided the data into respective categorizes. With the train model you could use your validation dataset to check if given the feature if the model will be placed into the right category. This makes this specific algorithm rather simple yet sophisticate, it tries to match the data set into the respective region. The data could be very flexible

5.3 BernoulliNB/MultinomialNB

The Naive Bayes family of statistical algorithms are some of the most commonly used algorithms in text classification. Overall it provided a variety of technique for analysis and type that fits different data. One of the method considered is Multinomial Naive Bayes (MNB) with a huge advantage, due to the small data set it still allows optimal classification Since the data is relatively small tens of thousands of sample as oppose

Naive Bayes is based on Bayes's Theorem which tries to calculate complex probability base on any of the given features such that how likely is it in certain category given these feature exist. So we're calculating the probability of each category probability given the text and then pick the highest probable genre to be the solution.A text sample will have to contain information about the probabilities of certain words within the texts of a given category from the training set. So that the algorithm can compute the likelihood of that plot summary is belonging to a certain genre.

5.4 Gradient Descent

is the most common optimization algorithm in machine learning and deep learning but not quite classification. It takes into account on each iteration what the changes are by taking the derivative of the function. With each step you you tweak the value in the opposite direction of the invalid data in order to achieve the minimum possible in order to classify the model.

5.5 KNN

KNN algorithm is one of the simplest classification algorithm it focus on learning base on new and old data set. It reference a pre-existing database of information by finding the distances between a category you are looking for and all the examples in the database. Looking for the most similar of the data base on user specified set number then determine base on those set which classification has the most vote or the average. This will be quick to compare the new data set from book summary so was included to compare to ensure greater accuracy.

5.6 LTSM

Long Short Term Memory networks are a special kind of RNN its one of the more human like learning system. It has the ability to form long term dependency and reference short term example model before making final classification. In order to make as data set grows it remember the initial training example and have that impact the results. All recurrent neural networks have the form of a chain of repeating modules of neural network sort of like recursive model where is output is then fed back into the training data. It takes a long time but usually yield the best result

6 Experimental Results

6.1 Logistic Regression

	precision	recall	f1-score	support
0	0.40	0.70	0.51	1200
1	0.23	0.48	0.31	540
2	0.37	0.46	0.41	358
3	0.56	0.56	0.56	2320
4	0.32	0.58	0.41	663
5	0.43	0.72	0.54	751
6	0.63	0.66	0.65	3626
7	0.19	0.42	0.26	343
8	0.21	0.42	0.28	274
9	0.04	0.02	0.03	174
10	0.14	0.39	0.21	150
11	0.40	0.75	0.52	819
12	0.33	0.54	0.41	272
13	0.13	0.37	0.19	259
14	0.28	0.48	0.35	773
15	0.31	0.64	0.42	338
16	0.12	0.06	0.08	106
17	0.26	0.59	0.36	835
18	0.23	0.65	0.34	141
19	0.31	0.71	0.43	155
micro avg	0.39	0.59	0.47	14097
macro avg	0.29	0.51	0.36	14097
weighted avg	0.43	0.59	0.49	14097
samples avg	0.42	0.61	0.47	14097

6.2 LinearSVC

	precision	recall	f1-score	support
0	0.62	0.41	0.49	1200
1	0.48	0.15	0.23	540
2	0.73	0.23	0.35	358
3	0.64	0.47	0.54	2320
4	0.44	0.14	0.22	663
5	0.82	0.58	0.68	751
6	0.65	0.62	0.64	3626
7	0.43	0.09	0.15	343
8	0.43	0.05	0.10	274
9	0.50	0.01	0.01	174
10	0.57	0.03	0.05	150
11	0.72	0.46	0.56	819
12	0.68	0.23	0.35	272
13	0.32	0.03	0.05	259
14	0.39	0.13	0.20	773
15	0.60	0.28	0.38	338
16	0.00	0.00	0.00	106
17	0.39	0.12	0.19	835
18	0.52	0.10	0.17	141
19	0.71	0.31	0.43	155
micro avg	0.63	0.38	0.48	14097
macro avg	0.53	0.22	0.29	14097
weighted avg	0.59	0.38	0.44	14097
samples avg	0.52	0.42	0.44	14097

6.3 Naive Bayes

	precision	recall	f1-score	support
0	0.09	0.00	0.00	1200
1	0.00	0.00	0.00	540
2	0.00	0.00	0.00	358
3	0.83	0.03	0.06	2320
4	0.05	0.00	0.00	663
5	0.09	0.00	0.01	751
6	0.77	0.36	0.49	3626
7	0.12	0.01	0.01	343
8	0.00	0.00	0.00	274
9	0.00	0.00	0.00	174
10	0.00	0.00	0.00	150
11	0.05	0.00	0.00	819
12	0.00	0.00	0.00	272
13	0.00	0.00	0.00	259
14	0.00	0.00	0.00	773
15	0.00	0.00	0.00	338
16	0.00	0.00	0.00	106
17	0.11	0.00	0.00	835
18	0.07	0.01	0.01	141
19	0.00	0.00	0.00	155
micro avg	0.67	0.10	0.17	14097
macro avg	0.11	0.02	0.03	14097
weighted avg	0.36	0.10	0.14	14097
samples avg	0.16	0.11	0.13	14097

	precision	recall	f1-score	support
0	0.53	0.31	0.39	1200
1	0.23	0.07	0.10	540
2	0.11	0.01	0.02	358
3	0.65	0.41	0.50	2320
4	0.18	0.03	0.06	663
5	0.62	0.36	0.46	751
6	0.69	0.63	0.66	3626
7	0.11	0.01	0.02	343
8	0.00	0.00	0.00	274
9	0.00	0.00	0.00	174
10	0.00	0.00	0.00	150
11	0.57	0.14	0.23	819
12	0.20	0.03	0.05	272
13	0.00	0.00	0.00	259
14	0.25	0.08	0.12	773
15	0.26	0.03	0.06	338
16	0.00	0.00	0.00	106
17	0.20	0.05	0.08	835
18	0.06	0.01	0.01	141
19	0.00	0.00	0.00	155
micro avg	0.58	0.30	0.39	14097
macro avg	0.23	0.11	0.14	14097
weighted avg	0.45	0.30	0.34	14097
samples avg	0.40	0.33	0.34	14097

6.4 Gradient Descent

	precision	recall	f1-score	support
0	0.91	0.10	0.17	1200
1	0.00	0.00	0.00	540
2	0.87	0.06	0.10	358
3	0.83	0.18	0.30	2320
4	0.00	0.00	0.00	663
5	0.89	0.39	0.54	751
6	0.72	0.55	0.62	3626
7	0.00	0.00	0.00	343
8	0.00	0.00	0.00	274
9	0.00	0.00	0.00	174
10	0.00	0.00	0.00	150
11	0.83	0.13	0.22	819
12	0.00	0.00	0.00	272
13	0.00	0.00	0.00	259
14	0.00	0.00	0.00	773
15	0.67	0.01	0.01	338
16	0.00	0.00	0.00	106
17	0.00	0.00	0.00	835
18	0.00	0.00	0.00	141
19	1.00	0.01	0.01	155
micro avg	0.76	0.21	0.33	14097
macro avg	0.34	0.07	0.10	14097
weighted avg	0.54	0.21	0.27	14097
samples avg	0.34	0.24	0.28	14097

6.5 KNN

	precision	recall	f1-score	support
0	0.57	0.24	0.34	1200
1	0.44	0.09	0.14	540
2	0.73	0.20	0.32	358
3	0.50	0.30	0.38	2320
4	0.42	0.09	0.15	663
5	0.68	0.26	0.38	751
6	0.58	0.53	0.56	3626
7	0.52	0.08	0.15	343
8	0.40	0.05	0.09	274
9	1.00	0.01	0.02	174
10	0.30	0.02	0.04	150
11	0.67	0.23	0.34	819
12	0.69	0.15	0.24	272
13	0.27	0.02	0.03	259
14	0.28	0.06	0.10	773
15	0.51	0.13	0.21	338
16	0.25	0.01	0.02	106
17	0.26	0.06	0.10	835
18	0.32	0.05	0.09	141
19	0.55	0.10	0.17	155
micro avg	0.55	0.27	0.36	14097
macro avg	0.50	0.13	0.19	14097
weighted avg	0.52	0.27	0.32	14097
samples avg	0.38	0.29	0.32	14097

6.6 LTSM

TBD

7 Conclusion

LTSM provided the highest accuracy but takes the longest, in the case of the classification it may not be worthwhile for such low optimization. Given the process few of the previous model may yield effective solutions.

The aim of this project is the text classification for the movie and book genres. We then apply it to book with relatively effective and efficent

We can conclude that the Combined RNNs is the best model in this classification task because it combine the learning experience and all historical value

8 Future Work

In the future, we would improve our models in several ways by tweaking various parameter. Changing the way we classify features. Or the modification of various loss function for the whole model to further verify lost. We should make a trade off, that is, a reasonable proportion for different part of loss along with optimization also taking into consideration of over-fitting. Focusing on LSTM structure with different sets of layers to use.

References

1. https://www.kaggle.com/jrobischon/wikipedia-movie-plots
2. https://www.kaggle.com/ymaricar/cmu-book-summary-dataset
3. https://www.kaggle.com/rounakbanik/the-movies-dataset?select=movies_metadata.csv
4. https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset