Machine Learning for Movie Genre Classification

Jiejun Lu, Hongxiang Qiu, Weidong Xu, Zeyu Zhao
{jiejunlu, qiu, wxu, zeyu_zhao}@g.harvard.edu

AC209B: Data Science 2

Spring 2018

Introduction

Nowadays, tons of new movies are released in US cinemas per year, with more on websites like Netflix and YouTube. And the number of new movies released each year is still growing. Therefore, movie genre classification, which helps people quickly find the movies they want to watch, becomes more and more important.

In this project, we implemented several machine learning models to classify genres of movies using their **overview description texts** and **poster images**. Our final model is a combined network with a Recurrent Neural Network (RNN) for text encoding and a Convolutional Neural Network (CNN) for image encoding.

Data

We scraped all 361,622 movies from TMDB. As data cleaning, we removed movies with short overview length (less than 10 words) since the majority of them are not enough to describe the movie. We also removed movies without available posters as we need posters in our final model. We are interested in recent movies, so after data cleaning, we only keep the latest 30,000 movies. There are 19 possible genres in total for movies in TMDB and **a movie can have multiple genres**. We plot the histogram of the 19 genres in Figure 1, and it's **very unbalanced**.

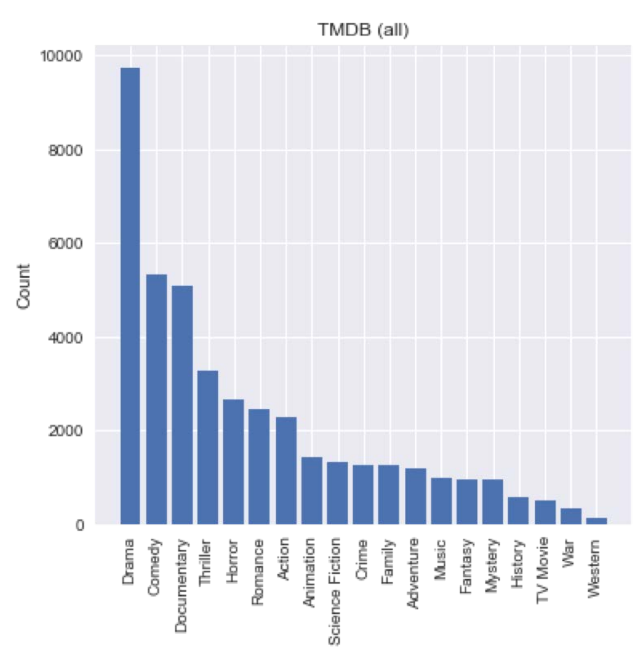

Figure 1. Counts of Genres

Figure 2 shows the overview description word cloud of *Crime* movies and *Romance* movies, as well as the posters of a *Documentary* movie and an *Animation* movie. Their patterns (both word clouds and images) are pretty different. So we think both overview description and posters are good indicators for genres.

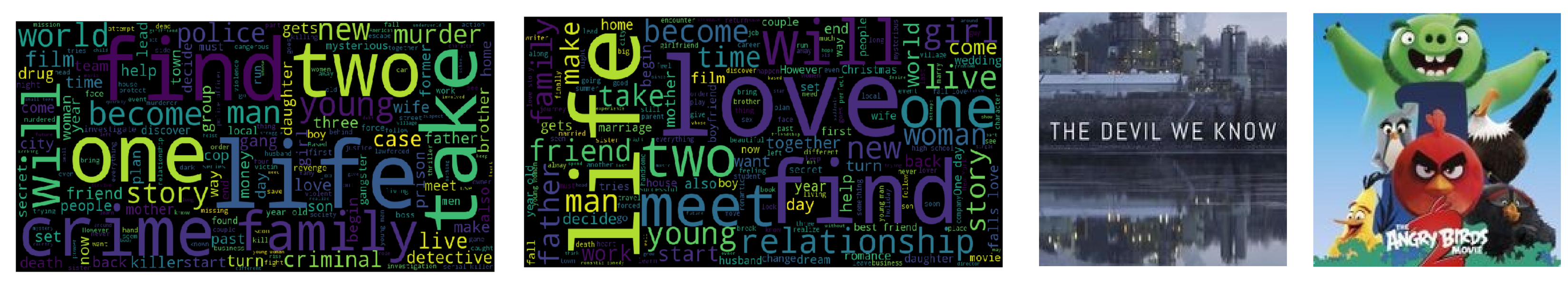

Figure 2. Word Cloud and Posters of Movies with Different Genres

Conventional Models

As baselines we have implemented several conventional models for text classification. We get vector representation of overview texts and use 19 classifiers to predict the genres of the movies. Our approach is also called *binary-relevance method* in multi-label learning.

For vector representation, we have tried bag-of-words, TFIDF [1], and word2vec [2] embeddings. For classifier, we have tried support vector machine (SVM) and multinomial naïve Bayes (NB).

RNN on Overview Texts

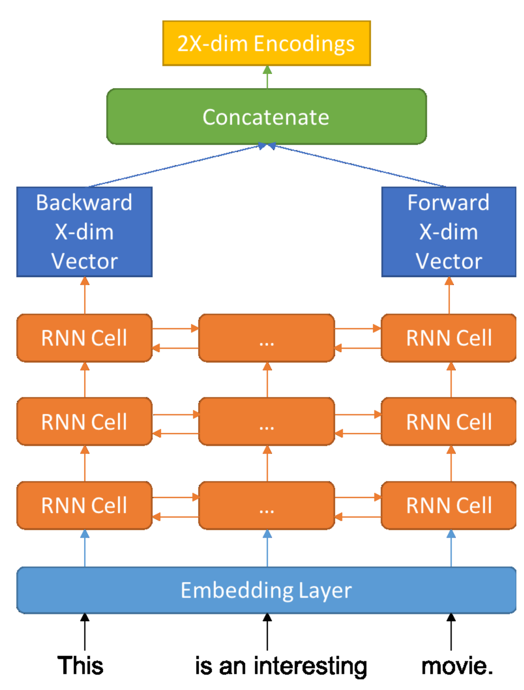

Figure 3. RNN Encoder

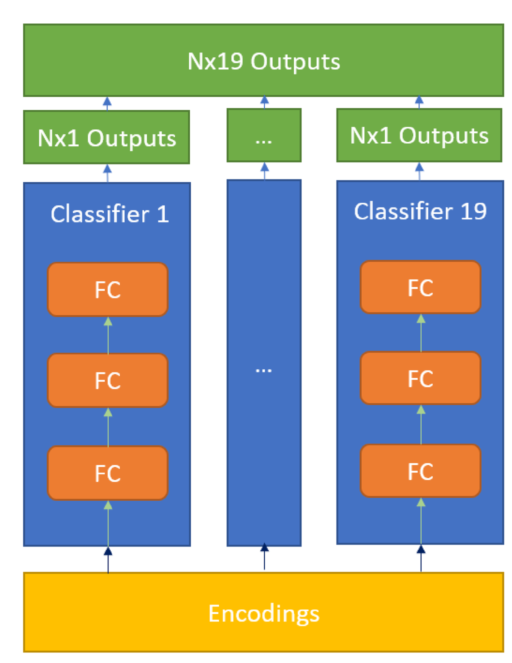

Figure 4. Classifier

Figure 3 and 4 shows our RNN architecture for encoding and classification. We tried both GRU and LSTM as RNN cell in Figure 3.

CNN on Poster Images

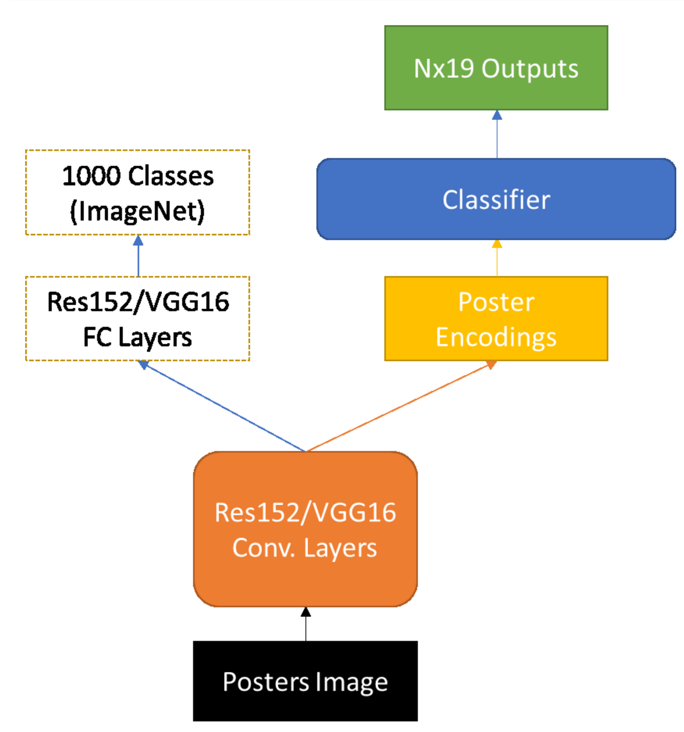

Figure 5. CNN Architecture

We use transfer learning for CNN. Our base network [3, 4] is trained on ImageNet. We only take the Convolution layers of the base network.

Figure 5 shows our CNN architecture for encoding and classification. The classifier has the same architecture with the one in Figure 4.

Combined Network

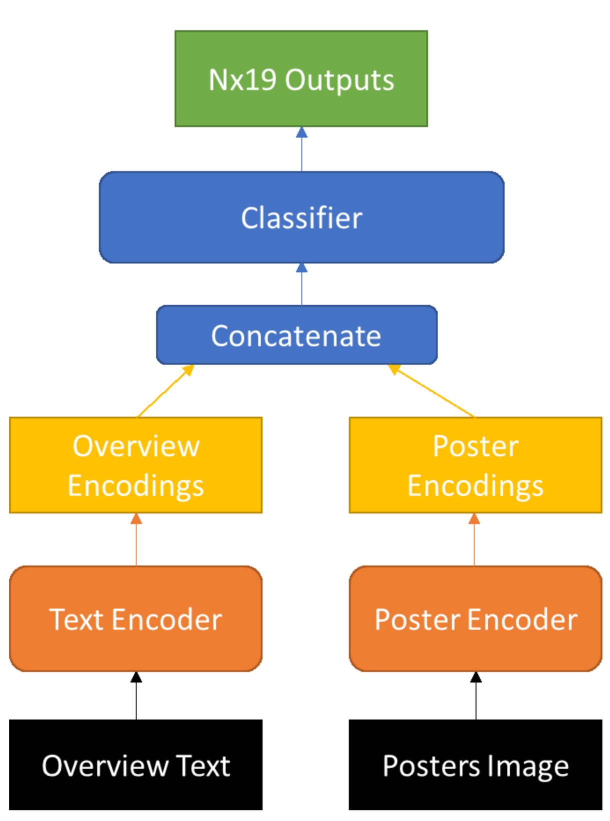

Figure 6. Combined Network

We take the best RNN encoder for overviews and the best CNN encoder for posters. Concatenate the encodings and only train a new classifier (freeze weights for the encoders).

Figure 6 shows our combined network (final model). The classifier has the same architecture with the one in Figure 4.

Results and Comparison

We report both accuracy and the micro averaged F_1-score [5] for all 4 types of models we described above on the **test set**. Due to the space limitation, we only list the best result for each type of the models.

	F_1-score	Accuracy
SVM + Bag-of-words	0.49	0.93
LSTM	0.51	0.93
Res152 based CNN	0.43	0.93
Combined Network	**0.52**	**0.94**

F_1-score is a better metric than accuracy. It's not surprising that our combined network yields the best F_1-score since this network takes account of both overviews and posters.

Citations

[1] Rajaraman, A., & Ullman, J. (2011). In Mining of Massive Datasets. *Data Mining*, pp. 1-17.

[2] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, pp. 3111-3119.

[3] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*

[4] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778.

[5] Sorower, M. S. (2010). A literature survey on algorithms for multi-label learning. *Oregon State University, Corvallis*, 18