CS230

Multi-label Movie Genre Classification Using Multiple Modalities

David A. LeBaron
Department of Computer Science
Stanford University
dlebaron@stanford.edu

Abstract

This project explores an ensemble approach to the task of multi-label movie genre classification. The proposed model combines data from movie posters, video trailer previews, text plot summaries, and metadata using separate deep networks. Other than the text model, the ensemble predictions show improved accuracy over the other models, which suggests that there is room for more research on effective ways to combine multiple data modalities.

1 Introduction

The task of genre classification is an important problem in machine learning. Accurate classification can be applied to both cataloging and recommending content to users. In the case of movie genre classification, usually a single label is not sufficient to properly characterize each example, e.g. a film can be both Action and Thriller, or Romance and Comedy. This project will explore the multi-label classification task. Each movie input will include a text plot summary, a poster image, a sequence of 200 movie trailer frames, and metadata such as director, actor, etc. A separate network is trained on each modality – one final output layer combines the predictions from each network into a 13 dimensional vector of probabilities representing each possible genre.

2 Related work

Prior research has taken varying approaches to the task of genre classification. Rasheed and Shah[8] demonstrate that a combined audio-visual model performs well on the single-label task, using a shot-detection algorithm designed to detect important frames. Convolutional networks have made this process easier. Simões et al.[7] improve on this approach by training a CNN on each key-frame detected by the shot-detection algorithm, followed by an averaging over frames to return a classification vector. Wehrmann et al. improve on this work by fine-tuning a CNN pre-trained on the ImageNet dataset[11] for object detection, along with introducing an ensemble model in which several networks "vote" to produce final classifications. Wehrmann et al. [1] also describe a very effective 3D CNN, in which spatio-temporal features are learned on sequences of trailer frames. Chu and Guo [3] show that multi-label classification can be achieved on only posters, by implementing a deep CNN that performs state-of-the-art object detection. [4] Oramas et al. provide much of the prior work on multimodal fusion. While their work is in the field of genre classification for music, much of their findings have been applied in this project. They show that training separate models on text, audio, and cover art, followed by a blending step, is more effective then training all modalities at once. Cascante-Bonilla et al. [5] provide the state-of-the-art work for the multilabel movie classification task. They provide the most complete dataset to date, including poster images, trailer frames, and

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

plot summaries. They build a deep network for classification that learns on these separate modalities simultaneously. This project relates to this work most directly, as their compiled dataset was used for training and evaluation. However, instead of simultaneous training, this project attempted to leverage the findings of Oramast et al. and Wehrmann et al., using an ensemble blending step to produce the final predictions.

3 Dataset and Features

The dataset is the MovieScope Dataset [5] provided by Cascante-Bonilla et al., which includes 3449 training samples, 491 validation samples, and 987 testing samples. Each example represents a movie that has included metadata such as director and actors, an image representing a poster, 200 images sampled from the movie's trailer preview, and a text synopsis of the plot. Each example also includes a 13 dimensional vector of labels representing the possible genres: action, animation, biography, comedy, crime, drama, family, fantasy, horror, mystery, romance, sci-fi, and thriller. The dataset is very diverse and consists of movies from all over the world, spanning 66 countries, and 48 languages, including movies from 2, 399 different directors and 4, 100 different actors. The oldest movie in the dataset is from 1920 and the latest is from 2016.

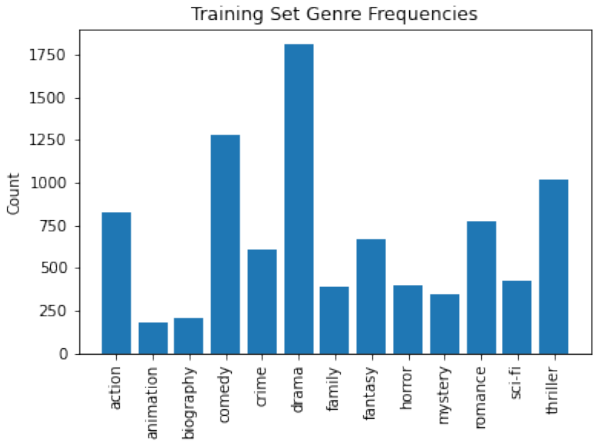

Fig (1) frequencies by genre

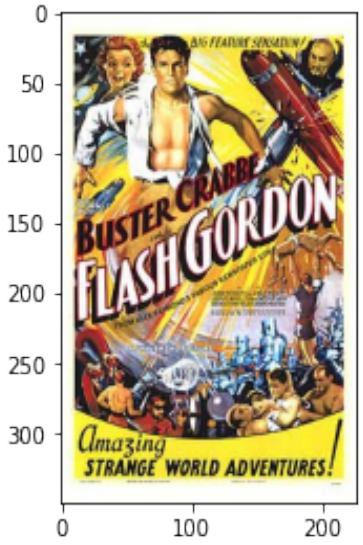

Fig (2) Example movie poster with labels: 'action', 'fantasy', 'sci-fi'

For preprocessing, images were normalized to 224X224x3. Text examples were truncated to length 3000 and transformed using the GloVe 42B CommonCrawl embedding [6]. Since class distribution is unbalanced, a weighting vector based on inverse frequency was included in the loss criterion to balance out contributions from each genre.

4 Methods

The classification model presented is an ensemble model where data from each modality is trained separately on the target, then the results are combined in a final output layer. All models use the mean Binary Cross Entropy loss:

$$l(x, y) = mean(L),$$
$$L = (l_1, ..., l_N)^T, l_n = -[p_c y_n * log x_n + (1 - y_n) * log(1 - x_n)]$$

Where p_c is the weight for a positive answer of genre c. Weights for each class were computed by the ratio of negative examples to positive examples, to reduce bias towards the more frequent genres. All models use a sigmoid activation layer to get the final probabilities for each genre.

For the poster images, a VGG-16 [10] CNN pre-trained on the ImageNet [11] dataset was used, and fine-tuned on the genre classification target by removing the last fully-connected layer and adding two fully-connected layers, while freezing the prior layers.

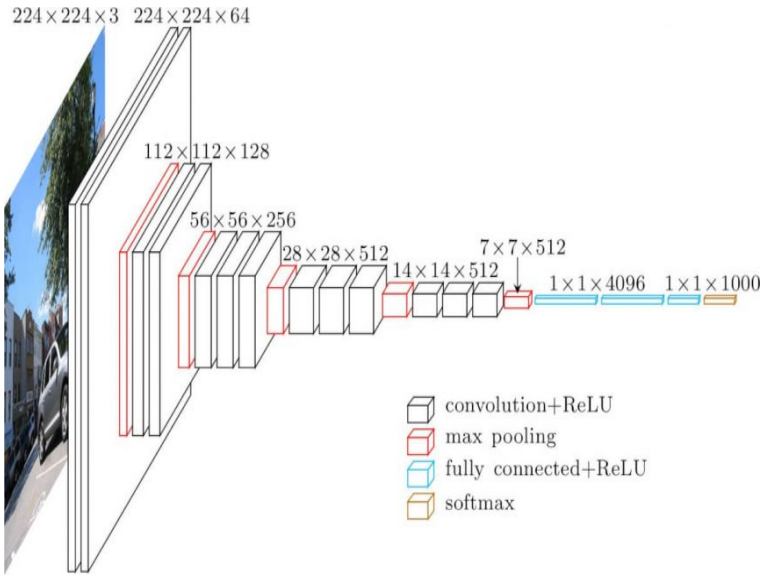

Fig (3) VGG-16 Architecture. For the poster representation task all weights up to the 4096 dense layer were frozen, and two new dense layers 4096x1000 and 1000x13 were appended to fine-tune on the classification task

For the movie trailer data, pre-trained VGG-16 was also used. To save on training time, the 200x4096 fully-connected outputs, provided by Moviescope[5] were used. Values were averaged over the 200 frames for each sample, to produce a 1x4096 vector like the poster data. These were fed through a 4096x1000 and 1000x13 dense layer as above, to get the outputs.

For the meta data, categorical features: `director_name`, `actor_1_name`, `actor_2_name`, `actor_3_name`, `language`, and `content_rating` were one-hot encoded into a length 11113 vector. These features were deemed to be most correlated with genre. A fully-connected network with 4 layers as used to produce outputs.

The plot summaries were truncated to max 3000 word length, then transformed into a 1x300 vector using the GloVe 42B CommonCrawl embedding [6]. Outputs were produced using a 300x50, and 50x13 fully-connected layers.

Outputs from each modality were then stacked and combined using a single fully-connected (13*4)x3 output layer.

5 Experiments/Results/Discussion

Models were trained using the BCE loss described above, with the Adam optimizer. Several hyper-parameters were tried, but no showed significant improvement over the default values. The chosen batch size of 64 resulted in the quickest convergence during training. Initial experiments did not use any weighting for positive samples in the loss function as described above, however this caused the model to pick the more common genres, such as 'Drama' most of the time. This problem was fixed by weighting positive examples of each class according to their frequency in the training dataset. Another problem that we encountered was that some of the models, such as the poster model, converged must faster during training, and the extra epochs caused the model to overfit to the training set. Adding an early stopping block in the training code, using the eval loss, allowed training to terminate if eval loss failed to decrease a few epochs in a row.

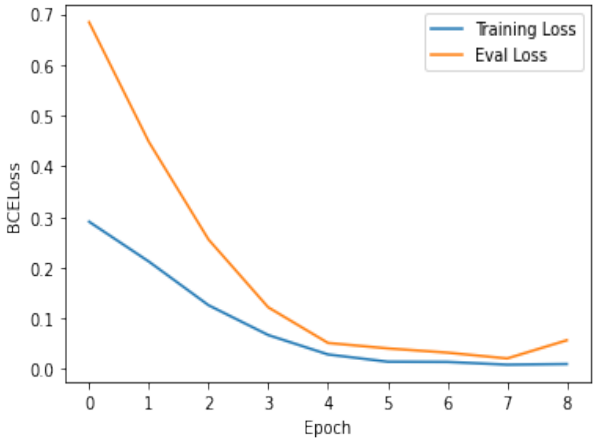

Fig (4) Learning curve of poster model with early stopping

The primary metric for evaluating performance on the test dataset is average precision, or the area under the precision-recall curve.

$$AP = \sum_n (R_n - R_{n-1}) P_n$$

Where R_n and P_n are the precision and recall at the nth threshold.

We use the Scikit-learn [12] implementation, which includes: the macro AP – the mean of the binary metrics giving equal weight to each class; the micro AP across samples which gives each sample-class pair an equal contribution to the overall metric; and the sample AP, which does not calculate a per-class measure and instead calculates the metric over the true and predicted classes for each sample in the evaluation data, returning their weighted average. The macro AP ensures that the algorithm performs well across all categories, even for those that have less training samples or are more difficult to predict. The micro AP ensures that we obtain overall good results across all samples. The sample AP is an individual evaluation in which each sample is computed separately from the whole set, and then returns an averaged score.

	micro	macro	samples
Poster	0.520138	0.446334	0.657156
Metadata	0.494375	0.457449	0.628409
Video	0.589880	0.573914	0.696374
Text	**0.631661**	**0.619510**	**0.749681**
Combined	0.626956	0.564063	0.735774

Fig (5) Average Precision scores for each model.

While all models perform well, they do not quite reach human level (estimated to be .72 macro AP [5]). While we see overall improvement over each modality in the ensemble model, the text model is the clear winner. This suggests several takeaways: that plot summaries are extremely informative in predicting genre, that there potential improvements to be made in the other modalities, as well as the ensemble approach. Particularly, a random forest model as suggested by [5] would likely do better on the metadata, due to the sparseness of the categorical features. I would also like to explore the convolutions through time approach on the video frame data, as shown by [2].

It is also important to explore qualitative examples to showcase the limitations of certain types of data, and how using multiple modalities can help improve accuracy.

Fig (6) Test example "Gravity". True labels are 'drama', 'fantasy', 'sci-fi', 'thriller'.

Figure 6 shows a movie example where the poster model benefits from additional information. The dark lighting and close-up, partially obscured face suggest something scary or mysterious – the poster model predicts 'horror', 'mystery', 'thriller', of which only 'thriller' is correct. The ensemble model, with the benefit of the plot summary, trailer frames, and metadata, more correctly predicts 'action', 'drama', 'fantasy', 'sci-fi', 'thriller'.

6 Conclusion/Future Work

In this project, I explored multiple model approaches using different types of data, as well as an ensemble method for combining them, for the task of multi-label movie genre classification. While all the models performed well, the results show that text-based models on human-written plot summaries are the highest performing. This is likely because a good plot summary must give the reader a good idea of the genre of the movie. Examination of a few samples show that including multiple modalities show promise in improving overall accuracy. Future improvements would be to investigate better approaches for combining multiple modalities, and applying a convolutions-through-time approach to the video trailers dataset.

7 Contributions

David LeBaron did the initial literature review, collected and cleaned the dataset, wrote code for model training and analysis, and wrote this report.

References

[1] J. Wehrmann, R. C. Barros, G. S. Simões, T. S. Paula and D. D. Ruiz, "(Deep) Learning from Frames," 2016 5th Brazilian Conference on Intelligent Systems (BRACIS), 2016, pp. 1-6, doi: 10.1109/BRACIS.2016.012.

[2] Jônatas Wehrmann, Rodrigo C. Barros, Movie genre classification: A multi-label approach based on convolutions through time, Applied Soft Computing, Volume 61, 2017

[3] Wei-Ta Chu and Hung-Jui Guo. 2017. Movie Genre Classification based on Poster Images with Deep Neural Networks. In Proceedings of the Workshop on Multimodal Understanding of Social, Affective and Subjective Attributes (MUSA2 '17). Association for Computing Machinery, New York, NY, USA, 39–45.

[4] Oramas, S., et al. (2018). Multimodal Deep Learning for Music Genre Classification. Transactions of the International Society for Music Information Retrieval

[5] Paola Cascante-Bonilla and Kalpathy Sitaraman and Mengjia Luo and Vicente Ordonez, Moviescope: Large-scale Analysis of Movies using Multiple Modalities. 2019.

[6] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation.

[7] G. S. Simões, J. Wehrmann, R. C. Barros and D. D. Ruiz, "Movie genre classification with Convolutional Neural Networks," 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 259-266, doi: 10.1109/IJCNN.2016.7727207.

[8] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews," 2002 International Conference on Pattern Recognition, 2002, pp. 1086-1089 vol.2, doi: 10.1109/ICPR.2002.1048494.

[9] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems 32 (pp. 8024–8035). Curran Associates, Inc. Retrieved from http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[10] Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014.

[11] Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

[12] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.