Context of Experience – MediaEval submission of ITEC / AAU

Polyxeni Sgouroglou, Tarek Markus Abdel Aziz, Mathias Lux
Klagenfurt University
Universitätsstrasse 65-67
Klagenfurt, Austria
{psgourog,tabdelaz}@edu.aau.at, mlux@itec.aau.at

ABSTRACT

People want to be entertained. However, context influences what people actually find entertaining. The *MultimediaEval 2016 Context of Experience Task* [2] focuses on automated methods to find the right content for a specific viewing situation, or more specifically, which movies are good to watch in an airplane. In this paper we present our approach to automatically suggesting movie from a list of possible ones by means of visual data as well as meta data.

1. INTRODUCTION

Movies for entertainment are big business. Worldwide TV and video revenue in 2015 is estimated with 286.17 billion USD[1]. With new shows, series and movies every year, there is a huge library of content to choose from. Especially in the confined space of an airplane seat and for the duration of a long distance flight, video entertainment is well received by passengers. So many companies offer on-board entertainment systems, where passengers can choose from multiple videos to entertain themselves without disturbing other passengers. While there is of course no one-fits-all solution, the general hypothesis of the task is, that some videos are better suited for watching on an airplane than others. Of course there are many different factors that can influence such a decision, ie. if a movie is still watchable on a small, low contrast screen, or if there are scenes which are potentially offending to neighboring passengers.

2. OUR APPROACH

While distinguishing positive from negative reviews for films is a quite easy process for humans, it appears that determining the suitability of a film for watching on an airplane, is even for humans a non trivial task. It becomes even harder when humans are asked to decide about the suitability of films, while being out of the specific context. For a human who remains out of the airplane context it is rather difficult to visualize the exact emotional impact of a film while watched during a flight. Mood, different tastes, stress levels during flight, anxiety problems are only some factors that could influence a passenger's decision. Hence, lists proposed by websites about what should or should not be watched on an airplane remain controversial. However, the majority of passengers share the same goal, which is to pleasantly pass time. Therefore, there are some characteristics based on common sense that intuitively make a film unsuitable for watching on an airplane for the majority of the passengers including films about airplane crashes, very violent ones, those with a high level of nudity, etc.

For the *MultimediaEval 2016 Context of Experience Task* we submitted four runs, whereas the first two are visual only runs and the latter two investigate text features. For the text part of our experiments we choose to work with meta data and more specifically with plots, synopses and plot keywords of the films, since we consider them as the more descriptive and concise types of meta data for deciding the suitability in our watching situation. The intuition we have is that other types of meta data such as genre, language etc. are not sufficient to characterize a film as unsuitable for watching on an airplane.

2.1 Classification based on visual features

We used two different visual components, which are posters and trailers, for the classification of a movie. The first run – Run 1 – uses only posters, and the second run – Run 2 – uses only trailers. For both runs we used the same machine learning techniques.

Most people determine the genre or other special features of a movie by looking at a poster. For instance, red fonts are sometimes a common hint that the movie includes horrifying or bloody scenes. These derived features play also an important role in the movie selection in air planes, because to every movie title there is also presented the belonging poster. The best way to develop an algorithm that determine on a poster, whether it is good or not, is to use machine learning techniques.

We decided to use *deep learning*. It is usually employed for large datasets, but our development set was too small. Therefore, we extracted only the vectors of the last hidden layer from a pre-trained deep neural network model, which can then be used as input to a separate machine learning system. [4]

We used the deep learning framework *Caffe* [1] devel-

[1] https://www.statista.com/statistics/259985/global-filmed-entertainment-revenue/, last visited 2016-09-27

MediaEval 2016 Workshop, Oct. 20-21, 2016, Hilversum, Netherlands

oped by the *Berkeley Vision Learning Center*. It can be used to load one of the pretrained models, which are provided by community members and researchers. We used the *BLVC GoogLeNet model* [3]. It has 22-layers and is trained on the dataset *ImageNet-2014* to detect 1 000 different classes of images. For each poster from the development set we extracted a vector with 1024 dimensions from the layer *pool5/7x7_s1*. It is the last hidden layer, and contains all high-level processing information which were created by the network.

As classifier we build a Support Vector Machine (SVM) with *scikit-learn*[2], a free software machine learning library. The advantages of SVMs are that they are effective in high dimensional spaces and in cases where the number of input-vectors is smaller than the number of dimensions, like in our situation. We used a linear kernel function, and the soft margin parameter C was set to 10. The result is Run 1.

We performed something similar with the trailers. From every trailer we extracted 200 frames with *ffmpeg*[3], a software for handling multimedia data and streams. We also extracted with the *BLVC GoogLeNet model* the vectors from the last hidden layer and we concatenated all 200 vectors of a trailer. The result was a vector with 204,800 dimensions. We trained with these vectors from the development set the SVM afterwards we classified the vectors from the test set. The result is Run 2.

2.2 Text based classification

We first obtain the plots of the baseline text data given by the organizers by the XML files for the whole dataset of 318 films. We parse the XML files and access the title and the plot of each film. Then we perform text processing on the plots by casefolding, tokenization, non-alphanumeric characters removal, stopping based on Google's stop words list to reduce computational and space complexity and stemming with the *PorterStemmer* of the Natural Language Toolkit (nltk)[4]. Finally, we store the XML plot tokens for further use. After preprocessing and feature vectorization the training corpus contains 1 972 distinct terms.

In both runs we employ a two-step classification. In the first step the text features are used to determine with a Naive Bayes classifier if movies are good to watch on an airplane or not. In case of a positive match, ie. the film is classified as being good to watch on an airplane, we compare the terms of the text features to a list of – in our opinion intuitive – terms, that make movies unsuitable for being watched on an airplane: {*airplane crash, airplane attack, hijack, hijacking, air force one, bomb, terrorist, kidnap, abuse, fascism*}. If the text features of a positive example match terms of this list twice, the classification result is changed to the negative example class. This basically allows us to focus more on precision than on recall and to reduce the number of false positive hits. Experiments based on the development data set have shown that this significantly improves precision, while reducing recall only marginally.

Our first text based run – Run 3 – contains predictions based on the baseline text features, expanded by a set of text features we obtained from relevant web pages. The number of features used is 5 000 ordered by term frequency. The first 1 972 are features extracted by the baseline tokenized XML plots, while the rest 3 028 are features extracted from our downloaded, preprocessed and tokenized web results. For creating features the *CountVectorizer* – scikit-learn's bag of words tool – is used. As analyzer parameter we use unigrams. As classification method we use Naive Bayes for multinomial models. In the final prediction vector if a film that is already classified as suitable contains at least two of the list's terms, we change the classification to unsuitable for watching on an airplane.

Our second text based run – Run 4 – is based on the 1972 baseline text features extracted, while the rest 3028 features are obtained from plot keywords and synopses of Internet Movie Database (IMDb)[5]. All the other characteristics remain as in Run 1.

3. RESULTS & DISCUSSION

In the ground truth 137 out of 225 movies were classified as suitable for watching on an airplane (positive examples). The remaining 88 were negative examples. Table 1 shows the results of our runs. Classifying all as positive would theoretically lead to a precision of 0.6089 and a recall of 1.0. In that sense our approach focusing on minimizing the number of false positives was not well chosen in terms of evaluation numbers. However, the meta data based runs (3 and 4) as well as the poster based run (1) are better than the naive classifier. Surprisingly enough the visual only runs perform comparably well in relation to the meta data based ones. The poster itself – with the given method – seems to carry enough information for classification.

Table 1: Results of the submitted runs giving true and false positives and negatives, precision (P), recall (R) and F1.

Run	TP	FP	TN	FN	P	R	F1
1	87	52	35	49	0.6259	0.6397	0.6327
2	92	60	27	44	0.6053	0.6765	0.6389
3	92	55	32	44	0.6259	0.6765	0.6502
4	88	54	33	48	0.6197	0.6471	0.6331

4. CONCLUSIONS

In this work we have applied convolutional neural networks as well as meta data based methods for classification. Although the ground truth leans towards positive examples and our approach focuses on minimizing false negatives we think the chosen methods provide interesting results, especially when considering, that the applied methods are well known approaches, ie. Naive Bayes Classifiers, SVMs, CNNs, TF*IDF, etc., not tuned to the use case besides the list of inappropriate concepts and can surely be extended to better fit the use case.

5. REFERENCES

[1] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[2] http://scikit-learn.org/stable/, last visited 2016-10-06

[3] https://ffmpeg.org/, last visited 2016-10-06

[4] http://www.nltk.org/, last visited 2016-10-06

[5] http://www.imdb.com/, last visited 2016-09-29

[2] M. Riegler, , C. Spampinato, M. Larson, P. Halvorsen, and C. Griwodz. The MediaEval 2016 context of experience task: Recommending videos suiting a watching situation. In *Proceedings of the MediaEval 2016 Workshop*, 2016.

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014.

[4] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li. Deep learning for content-based image retrieval: A comprehensive study. In *Proceedings of the 22nd ACM International Conference on Multimedia*, MM '14, pages 157–166, New York, NY, USA, 2014. ACM.