The IITB Predicting Media Interestingness System for MediaEval 2017

Jayneel Parekh
Indian Institute of Technology,
Bombay, India
jayneelparekh@gmail.com

Harshvardhan Tibrewal
Indian Institute of Technology,
Bombay, India
hrtibrewal@gmail.com

Sanjeel Parekh
Technicolor, Cesson Sévigné,
France
sanjeelparekh@gmail.com

ABSTRACT

This paper describes the system developed by team IITB for MediaEval 2017 Predicting Media Interestingness Task. We propose a new method of training based on pairwise comparisons between frames of a trailer. The algorithm gave very promising results on the development set but did not impress on test set. Our highest achieved MAP@10 on test set is 0.0911 (Image subtask) and 0.0525 (Video subtask), based on a systems submitted last year ([4, 6]).

1. INTRODUCTION

The MediaEval 2017 Predicting Media Interestingness Task [2] deals with automatic selection of images and/or video segments according to their interestingness to a common viewer. We only use the visual content and no additional metadata.

Previous systems on this task discuss in detail several relevant inherent problems. Further, they also point towards the usefulness of CNN features: in particular, they report features from AlexNet's fc7 layer performing reasonably well with simple classifiers [4, 6]. We believe a key shortcoming of the previous approaches is that they attempt to tag images interesting/non-interesting in a global context whereas the task inherently expects to classify images in a local context (trailer-wise). Our system tries to take this aspect into account by training a classifier on pairwise comparisons of frames from same trailer.

2. SYSTEM DESCRIPTION

2.1 Pre-processing

Given the training data feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$ consisting of N examples, each described by a F-dimensional vector, we first standardize it and apply principal component analysis (PCA) to reduce its dimensionality. The transformed feature matrix $\mathbf{Z} = (z_i)_i \in \mathbb{R}^{N \times M}$ is used to experiment with various classifiers. Here M depends on the number of top eigenvalues we wish to consider.

For our system we use AlexNets's fc7 [3] features provided for image subtask and C3D [8] features provided for video subtask. Each feature vector has a dimension of 4096. After performing PCA we reduce the dimension to 200. Thus $\mathbf{Z}$ is a $\mathbb{R}^{N \times 200}$ matrix in our system.

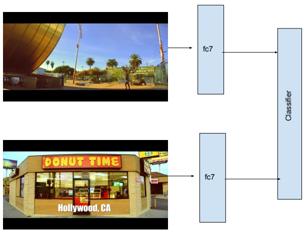

Figure 1: Pairwise comparison based Training: Concatenated images features (fc7) from same trailer are fed to the classifier and it learns to predict the more interesting image

2.2 Training

We adopted the following two methods for training:

1. Feed every frame/video's feature vector to the classifier where it learns to predict the interestingness label of the frame as in [4]
2. For each trailer we consider all possible pairs of its frames/videos and feed the corresponding concatenated feature vectors to the classifier. The classifier learns to predict which one of the two frames/videos is more interesting.

For the *second training method*, pairwise comparisons are made . First, from each trailer, we generate all possible pairs of frames. This ensures that only frames/videos of the same trailer are being compared. Considering T trailers having n_i number of frames/videos in them, we get $N_1 = \sum_{i=1}^{i=T} \binom{n_i}{2}$ pairs. Representation of each pair is done by concatenating the feature vectors of each frame/video. The feature vector of each being of size M, after concatenating we get final feature vector of size $2M$. This procedure yields a feature matrix $\mathbf{Z}_{new} \in \mathbb{R}^{N_1 \times 2M}$. Output labels for an ordered pair of frames/videos (I_1, I_2) is assigned as follows:

$$y = \begin{cases} 1, & I_1 \text{ is more interesting than } I_2 \\ 0, & I_2 \text{ is more interesting than } I_1 \end{cases} \tag{1}$$

2.3 Prediction

For the first two runs which are based on [4], [6], we have used different classifiers. Support vector machines (SVM) with rbf kernel (run1) and logistic regression with ℓ_1 penalty (run2). We now describe the prediction algorithm for our new approach.

MediaEval'17, 13-15 September 2017, Dublin, Ireland.

Ranking of the frames/videos according to their interestingness in a particular trailer is determined from the predicted results of all the pairwise comparisons by generating penalty scores s_i for each of them and ordering them from lowest to highest with lowest corresponding to most interesting frame/video. The scores are determined using the following algorithm (referred as P1):

1. Initialize the penalty scores $s_i = 0$ for each i
2. Iterate over results of all pairwise comparisons: for each pair indexed by $\{k, l\}$, let $r(k, l)$ denote the prediction of classifier. The following update is performed:

   $$s_u = s_u + |\Pr\{r(k,l) = 1\} - \Pr\{r(k,l) = 0\}|$$

   where u denotes the index of less interesting frame/video predicted, $\Pr\{.\}$ denotes the probability and $|.|$ the absolute value

This essentially increases the penalty score for the less interesting according to the confidence the classifier has in its prediction. The confidence value of the classifier for a given pair is treated as absolute difference between $\Pr\{r(k,l) = 1\}$ and $\Pr\{r(k,l) = 0\}$. We also try a variant of the above algorithm in one of our runs wherein the update equation is: $s_u = s_u + 1$ (referred as P2)

Interestingness classification: We opt for a simple method for binary classification of each image as interesting or not: We classify the top 12% ranked images as interesting. We chose top 12% images as it's slightly higher than the average number of interesting images, which is about 9%. It's important to note that since we generate ranking of frames, choosing only top 12% images has no particular significance as the official metric remains unaffected by it.

3. EXPERIMENTAL VALIDATION

The training dataset consisted of 7396 frames extracted from 78 movie trailers with about 392,000 pairs of frames, while the test data consisted of 2435 frames extracted from 30 movie trailers. [2] gives complete information about the preparation of the dataset. Scikit-learn [5] was used to implement and test various configurations.

3.1 Results and Discussion

Our results on the development set for various approaches are given in Table 1. The run submission results are given in Table 2. The tables gives the mean average precision (MAP) - the official metric MAP@10, of different runs corresponding to the method of training and the classifier used.

Development Set

We experimented with the CNN features provided and used PCA to bring down the number of dimensions to 200. Additionally, we used a non pairwise (NP) and a pairwise strategy (P1, P2) for training and prediction as described in previous section. These methods were used to train SVM (rbf kernel) [7], logistic regression with l1-penalty (LR-l1) [9]. These decisions were taken following inferences of previous results [4, 6]. We split the development set into the training set (62 videos) and cross validation set (16 videos). We calculated MAP@10 on the validation set. Accordingly we tested the model with several parameters and chose the model parameters giving best MAP@10 results. We found that the pairwise comparisons strategy was working better compared to non pairwise strategy. They gave a better MAP@10, which was aligned with our expectation. Logistic regression was giving better results as compared to SVM.

Run	Classifier	Subtask	MAP@10
1	NP + SVM-rbf	Image	0.094
2	NP + LR-l1	Image	0.144
3	P1 + LR-l1	Image	**0.179**
4	P2 + LR-l1	Image	0.178
5	NP + SVM-rbf	Video	0.088
6	NP + LR-l1	Video	0.092
7	P1 + LR-l1	Video	**0.109**
8	P2 + LR-l1	Video	0.108

Table 1: Results on development set

Run	Classifier	Subtask	MAP	MAP@10
1	NP + SVM-rbf	Image	0.1886	0.0500
2	NP + LR-l1	Image	**0.2570**	**0.0911**
3	P1 + LR-l1	Image	0.2038	0.0494
4	P2 + LR-l1	Image	0.2054	0.0521
5	NP + SVM-rbf	Video	**0.1795**	**0.0525**
6	NP + LR-l1	Video	0.1675	0.0445
7	P1 + LR-l1	Video	0.1700	0.0474
8	P2 + LR-l1	Video	0.1678	0.0445

Table 2: Run Submissions: MAP@10 (official metric)

Due to large number of pairs involved in training, we could not experiment with classifiers such as SVM in our presented approach (P1, P2) because of large training time. We experimented with the following classifiers. (1) Logistic regression with l2 penalty, (2) Random Forest, (3) Logistic regression with l1 penalty. (3) gave slightly better results than the other two and was the fastest in training, hence we went with logistic regression-l1 penalty as our classifier.

Test Set

However the results on the test set, were unexpected. Logistic regression using pairwise comparisons gave the best results on the development set for both the tasks. On the test set it isn't impressive where the best result is for non-pairwise logistic regression (Image subtask) and non-pairwise SVM-rbf kernel (Video subtask).

There could be various possible reasons for the discrepancy in the results on development and test set. (i) Viewing the classifier as a neural network, it may require more fine tuning of the weights of fc7 layer of AlexNet, or a more complex network instead of a single neuron so that it generalizes better. (ii) Though improbable, it's possible there are some discrepancies in the sources of development and test set which result in poor generalization.

4. CONCLUSIONS

In summary, we proposed a new system for interestingness prediction in images and videos. It essentially differs in the method of training based on pairwise comparisons of images. This helps in capturing interestingness of an image in a local context. Although our system gave impressive results on development set, it failed to perform well on the test set. Some improvements on current system can be improving its complexity or fine tuning the last layer of AlexNet for better input representation. The efficiency of the training can also be improved by selecting pairs more intelligently ([1]).

5. REFERENCES

[1] R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[2] C.-H. Demarty, M. Sjöberg, B. Ionescu, T.-T. Do, M. Gygli, and N. Q. Duong. MediaEval 2017 Predicting Media Interestingness Task. In *Proc. of the MediaEval 2017 Workshop, Dublin, Ireland, Sept. 13-15, 2017*, 2017.

[3] Y.-G. Jiang, Q. Dai, T. Mei, Y. Rui, and S.-F. Chang. Super fast event recognition in internet videos. *IEEE Transactions on Multimedia*, 17(8):1174–1186, 2015.

[4] J. Parekh and S. Parekh. The MLPBOON Predicting Media Interestingness System for MediaEval 2016. In *MediaEval*, 2016.

[5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[6] Y. Shen, C.-H. Demarty, and N. Q. Duong. Technicolor@ mediaeval 2016 predicting media interestingness task. In *MediaEval*, 2016.

[7] J. A. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[8] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[9] H.-F. Yu, F.-L. Huang, and C.-J. Lin. Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1-2):41–75, 2011.