A Case Study of Deep Learning-Based Multi-Modal Methods for Labeling the Presence of Questionable Content in Movie Trailers

Mahsa Shafaei* , Christos Smailis*, Ioannis A. Kakadiaris, Thamar Solorio
Department of Computer Science, University of Houston, Houston, TX, USA
mshafaei@uh.edu, csmailis@central.uh.edu, ikakadia@central.uh.edu, tsolorio@uh.edu
* These authors contributed equally to this work.

Abstract

In this work, we explore different approaches to combine modalities for the problem of automated age-suitability rating of movie trailers. First, we introduce a new dataset containing videos of movie trailers in English downloaded from IMDB and YouTube, along with their corresponding age-suitability rating labels. Secondly, we propose a multi-modal deep learning pipeline addressing the movie trailer age suitability rating problem. This is the first attempt to combine video, audio, and speech information for this problem, and our experimental results show that multi-modal approaches significantly outperform the best mono and bimodal models in this task.

1 Introduction

Movie trailers can be found in abundance throughout the web using services such as video streaming platforms. However, not all types of content in trailers are suitable for every audience. Specifically, movie trailers may contain explicit, aggressive, or violent content that may be harmful to the psyche of young viewers. Previous research has documented that some of the negative effects of mass media in young viewers include aggression and anxiety (Wilson, 2008; Chang and Bushman, 2019), as well as increasing the risk of sexual onset and alcohol and drug consumption, unwanted pregnancies, and sexually transmitted diseases (Strasburger, 1989).

The Advertising Administration of the Motion Picture Association of America ("MPAA") established guidelines for manually rating the age-suitability of movie trailers (Motion Picture Association). The rating of movie trailers is independent of the rating of the movie itself, as a trailer includes only a short overview of the entire movie. Due to the time consuming nature, as well as the challenges to scale the MPAA rating process, automating the task is of practical value. Moreover, automating the task poses interesting challenges to multi-modal classification systems, as the source of the objectionable content can come from any, or the combination of, these sources: language (use of bad words or discussion of adult themes), images (graphic violent scenes, nudity, drug or alcohol use), and audio (loud noises and music score denoting suspenseful content). A successful rating approach should integrate evidence provided by the multiple modalities when making the predictions.

In this paper, we study the performance of different multi-modal deep learning methods, to automatically predict the MPAA age-suitability rating of movie trailers using cues from the video, audio, and text modalities. Our goal is to show the feasibility of automating the rating task, and in particular, the relevance of multimodal solutions. We explore the use of late fusion, feature concatenation fusion and Gated Multi-modal Unit (GMU) Fusion (Arevalo et al., 2017). Since the proposed pipeline does not use any type of metadata for trailers or movies, it can be easily extended to be applied to any type of online video content. The main contributions of this work are: (i) we introduce a new task in multi-modal classification; rating videos based on the MPAA rating metric for movie trailers; (ii) we introduce the Multi-modal Movie Trailer Rating (MM-Trailer) dataset that contains movie trailers and their corresponding MPAA tags, audio files, subtitles of the trailers, and the metadata of the target movie; and (iii) we demonstrate empirically that combining the different modalities yields significant improvements over the strongest monomodal model. Our results show that both, the GMU and late fusion approaches yield promising results.

2 Related Work

This work is related to four different areas, namely: (i) text classification, (ii) video classification, (iii)

Proceedings of Recent Advances in Natural Language Processing, pages 1297–1307
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_146

audio classification, and (iv) movie classififcation datasets.

Text Classification: In (Martinez et al., 2019), the authors proposed an RNN-based architecture for detecting violence in movies on a segment level as well as the full movie level, by using the movie's script. In Shafaei et al. (2019), the authors proposed an RNN-based architecture with an attention mechanism that jointly models the genre and the emotions in movie script to predict the MPAA rating of a full movie. The main difference between our work and aforementioned papers is that they only use scripts to predict the movie ratings (violence rating and MPAA ratings), while we employ various modalities (audio, video, and text) to predict if a trailer (not the entire movie) is appropriate or not for children. It should be noted that the rating schema is different for trailers compared to movies (details in Section 3), and movies are not freely available on the internet.

Video Classification: Early approaches, such as (Karpathy et al., 2014) on video classification using Deep Learning, explored the use of several temporal fusion methods for combining information from multiple consecutive video frames using features extracted from CNN architectures. The authors in (Donahue et al., 2015) introduced an end-to-end architecture based on a combination of CNNs used for feature extraction from RGB frames. The CNN features are then forwarded to an LSTM layer that models the temporal variation of frames. A different approach is followed in (Tran et al., 2015), namely 3D-CNN, where authors propose the use of a CNN variant that takes into account convolutions performed into both the spatial and temporal domains of a video. An expansion of the 3D-CNN approach was proposed by (Carreira and Zisserman, 2017), where the authors propose a two-stream 3D-CNN architecture for video classification. Again the two streams used as input RGB frame data and Optical flow images.

Audio Classification: In past research, several types of handcrafted feature extraction techniques have been proposed for the audio modality (Davis and Mermelstein, 1980; Geiger et al., 2013; Papakostas et al., 2017) with the ones being the most prominently used in the literature being Mel-frequency cepstral coefficients (MFCCs). However, recently several approaches have been proposed for combining audio features such as spectrogram information with deep learning architectures to perform audio classification (Papakostas et al., 2017; Hershey et al., 2017; Koutini et al., 2019). Audio has been explored as a modality for classifying movie content in several works such as (Rasheed and Shah, 2002; Hebbar et al., 2018). However, none of these methods has focused on the problem of movie trailer age-suitability rating.

Movie Classification Datasets: Several movie classification datasets have been proposed in the past. In (Demarty et al., 2014), the authors introduced MediaEval 2013 Violent Scene Detection, which provided annotations for detecting violent scenes in movies. In Constantin et al. (2020), the authors proposed an evaluation framework, for Violent Scenes Detection in Hollywood and YouTube videos along with a dataset (VSD96). Although these datasets are relevant to our work, they only cover the violence aspect and cannot address the problem of age-suitability rating (violence is only one of many aspects of age rating). In (Shafaei et al., 2019), the authors proposed a movie dataset focusing on the task of predicting the MPAA rating of the movie. However, the aforementioned dataset only includes movie scripts and corresponding metadata but does not include movie trailers or related age-suitability tags (As we mentioned earlier, the MPAA rating scheme is different for movies and trailer). In (Cascante-Bonilla et al., 2019), authors introduced Moviescope, a dataset for movie genre classification. Similarly, it does not include MPAA age-suitability rating labels for movie trailers.

3 Dataset

To the best of our knowledge, there is no previous trailer dataset with age-suitability rating. Thus, we assembled the multi-modal Movie Trailer dataset (MM-Trailer) [1] by collecting the rated trailers from the IMDB website and YouTube. Typically trailers are advertising movies soon to be released and shown in theaters before a movie starts. Rating in trailers is shown by a colored band (red, yellow, green) and a message that appears at the beginning of the trailer. The rating of the trailers adheres to the rating of the movie being shown in the theater. For instance, if the movie playing at the theater is rated as NC-17 (no one under 17 is recommended to watch this movie), the green band trailer that is advertised before this movie may not be appropriate for children even if the color is green. The

[1] https://ritual.uh.edu/RANLP2021/

Green Band Trailers	Red Band Trailers	Total Trailers
1,040	403	1,443

Table 1: Dataset statistics

yellow band is designed for trailers advertised on the internet, and it indicates that the corresponding trailer is suitable for "age-appropriate internet users" as visitors to sites are mainly adults. The last group of trailers are red band trailers; red color indicates the content is only appropriate for a "mature audience" or "restricted audience".

Since our goal is to design an automated system that is able to predict which movie trailers are not recommended for children, we define only two classes of trailers for the dataset:

1. **Green-band trailers:** this category includes (i) trailers with the message "all audiences", and (ii) green band trailers with "appropriate audience" whose associated movie is rated as G and PG.

2. **Red-band trailers:** all red-band trailers, these include restricted and mature audiences (not appropriate for children).

We also extracted separate audio files and trailer subtitles. Subtitles include narrator and actor speech. Some of the YouTube trailers include the video subtitle. For these cases, we pre-process the subtitles by removing timestamps to keep only words. For trailers that do not include a subtitle file, we use a python speech recognition tool (Zhang) to automatically generate the subtitle from the audio. Our dataset includes 11G of audio streams. For each trailer the audio file is a combination of background music and vocals together, so the duration of audio is the same as the duration of the trailer. The number of total words in all trailer scripts is equal to 1,478,139 (on average, there are 576 words per trailer). Note that 20,783 words of the vocabulary set are unique words. Table 1 shows the statistics of our dataset.

4 Methodology

Our goal is to predict the age-suitability rating for movie trailers following the guidelines of the Advertising Administration of MPAA for trailer rating. The problem is formulated as a binary classification task where trailers are labeled as either appropriate for all audiences (green-band trailers) or restricted audiences (red-band trailers). To achieve this goal, the Multi-modal Movie Trailer Rating (MMTR) system is proposed. Within this system, the trailers are modeled as a fusion of three modalities: subtitles, audio, and video of the trailers. We train Recurrent Neural Networks (RNNs) for subtitles and audio, and a combination of Convolutional Neural Networks (CNNs) with LSTM for video, as separate streams in order to extract a representation for each respective modality. Then, we combine all stream representations using a fusion module to take advantage of the cues coming from different modalities. Figure 1 shows the overall design for the system architecture.

Our approach is based on independently identifying the best individual modality model and then combining information from all three monomodal models (subtitle, audio, and video) through one of the following three fusion methods: (i) Gated Multi-modal Unit (GMU) (Arevalo et al., 2017), (ii) Feature Concatenation Fusion, or (iii) Late Fusion. All modules of the system are described in the following sections.

4.1 Text Stream

The subtitles of the trailers are a rich source of information. They can help in identifying the topic of the video content. Moreover, the presence of specific words in the dialogue can be a strong indicator for some types of sensitive content, while more subtle cues can be inferred from analyzing the entire transcript. To model the information originating from the subtitles, we feed them to the following modules:

BERT + Long Short-Term Memory (LSTM) with Attention: We use BERT (Devlin et al., 2018) to leverage the well-known power of transformer-based word representations. The word vectors are then passed to an LSTM layer to model the sequence of the words in order to extract the semantic information of the text. Afterwards, the resulting hidden representation of the LSTM is passed to an attention mechanism (Bahdanau et al., 2014) to find the importance of each word in the dialogue. Even though BERT has seen a series of improvements (RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019)), our goal in this paper is to present empirical evidence that a multi-modal approach can solve this task with acceptable performance, the specific contextualized embeddings used being of less relevance.

Emotion Vector: We expect to observe that

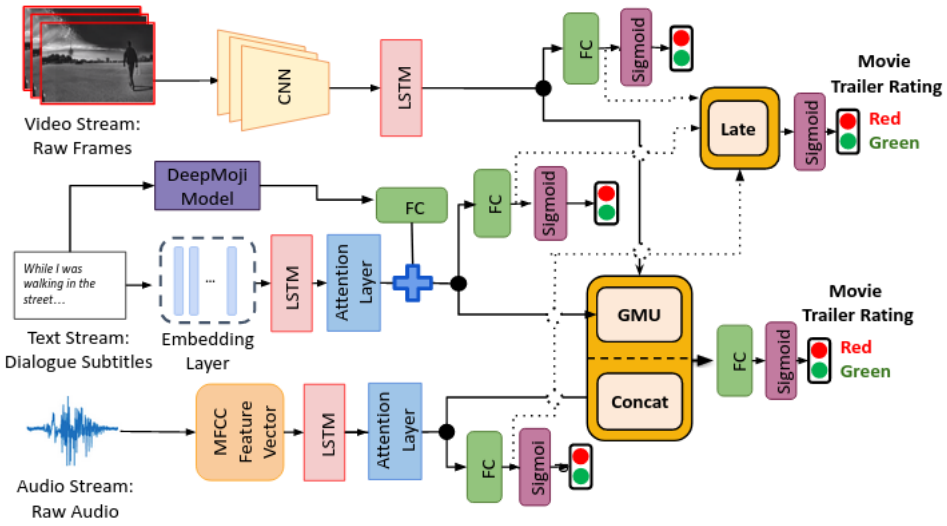

Figure 1: Overview of the system: (i) A video subtitle is transformed into a vector representation using an Embedding Layer and then forwarded to an LSTM network with an attention layer. We concatenate the output of the attention layer with the feature vector from the DeepMoji Model. (ii) A video volume is passed through a CNN-LSTM model that is used as a feature extractor, in order to obtain a single vector representation of the entire video. (iii) Raw audio signal from the video is represented as a sequence of MFCC feature vectors, passed to an LSTM layer. (iv) Lastly, information from all modalities is combined using one of the following fusion methods, namely Gated Multi-modal Unit (GMU), Late Fusion and Feature Concatenation Fusion, before labeling the age-suitability of each trailer. FC in the diagram stands for a fully connected layer.

strong negative emotions (fear, anger, sadness) correlate more with red band trailers. Similarly, positive emotions, such as joy, are more correlated with green band trailers. We made this assumption following research by (Shafaei et al., 2019) where they found promising results for using emotions in a movie rating task.

We model emotions with the use of the DeepMoji model (Felbo et al., 2017). This model was trained using 1.2 billion tweets with emojis to understand how language is used to express emotions. Recent work in abusive language detection shows promising results from using DeepMoji (Safi Samghabadi et al., 2019), thus it seems reasonable to expect good results in this task as well. To incorporate this model into our system, the last hidden layer representation of the pretrained model was used to transfer the text to emotional feature vectors. Finally, the emotion vector was concatenated with the output of the attention and the entire vector was passed to a fully connected layer to further fine-tune the joint representation.

4.2 Video Stream

The video modality is a rich source of visual and temporal cues that are useful for analyzing multimedia content. Specifically, in this task, video can help for modeling the objectionable content such as a depiction of nudity or bloody scenes and suggestive elements. To this end, in order to learn spatiotemporal video features, a CNN-LSTM model based on the works by (Donahue et al., 2015) and (Yue-Hei Ng et al., 2015) is adopted. Each video is sub-sampled to a fixed number of frames, evenly distributed across its duration to form a visual temporal sequence. The raw RGB frames are used as input to a CNN model. This CNN model produces a feature representation for spatial information within each frame. The output of the final pooling layer of the CNN is passed to an LSTM that models temporal dependencies between frames.

4.3 Audio Stream

The audio of the trailer can help the model to learn the genre and theme of the movie, and as a result, it is a powerful tool to distinguish red-band trailers from green-band ones. For example, *horror* and *thriller* movies (that usually include suspenseful music) are less likely to be suited for children. In addition to the music score, the emotion conveyed by the speakers' tone and pitch can provide relevant cues for rating the trailer. It should be noted that the entire audio is used in our model (the music and dialogue combined). To model the audio, the Mel Frequency Cepstral Coefficients (MFCC) are extracted from the audio stream. MFCCs are one of

the most common feature representations for audio classification (Andén and Mallat, 2011) and speech recognition tasks (Tiwari, 2010). The entire audio is divided to n chunks, $n \in \{10, 20, 50, 100\}$, then the MFCC feature vector is extracted for each chunck. Moreover, by performing averaging over the MFCC vector in each chunk, a fixed-length representation for the entire audio, regardless of its duration is obtained. The vector is then passed to an LSTM module to model the MFCC variations during the entire video. Lastly, by adding an attention mechanism, the model learns the importance of each audio chunk and feeds the weighted average of LSTM hidden representation to a fully connected layer that helps the model to be fine-tuned for the task.

4.4 Fusion

The goal of the fusion module is to learn to predict the rating of the trailer by integrating evidence from the video, audio and text modalities. We evaluate three established fusion methods in order to form a unified representation for each trailer.

Gated Multi-modal Unit (GMU): The GMU allows the model to learn an intermediate representation by combining the different modalities, where the gate neurons learn to decide the contribution of each modality to the intermediate representation. A great advantage of the GMU model is its ability to adjust the activation from each modality depending on the specific instance. This method is inspired by control flow in recurrent architectures. In RNN models, the recurrent units decide how much the current and previous evidence engage in building the current state. In GMUs, the activation function for building the output using different modalities is measured, in order to form a unified intermediate representation for all modalities.

The original GMU was successfully applied to a movie dataset of plot synopsis and movie posters to predict genre. In the original paper, the authors implemented a bimodal system (the equation is provided in the Appendix). We follow their formulation to extend the model to include three modalities using the straightforward approach discussed in their paper. The exact formulation is shown in Equation 1; where W_i, Y_i are learnable parameters, x_i is the feature vector for modality i and $[.,.]$ stands for concatenation.

$$
\begin{aligned}
h_1 &= tanh(W_1.x_1) \\
h_2 &= tanh(W_2.x_2) \\
h_3 &= tanh(W_3.x_3) \\
z1 &= \sigma(Y_1.[x_1, x_2, x_3]) \\
z2 &= \sigma(Y_2.[x_1, x_2, x_3]) \\
z3 &= \sigma(Y_3.[x_1, x_2, x_3]) \\
h &= z_1 * h_1 + z_2 * h_2 + z_3 * h_3
\end{aligned}
\tag{1}
$$

Feature Concatenation Fusion: One popular fusion method is generating a joint multi-modal representation through feature concatenation (Baltrušaitis et al., 2018) where the representation vectors of each modality are concatenated, and the unified representation is passed through multiple hidden layers or used directly for the prediction.

Late Fusion: Another vastly used fusion method is late fusion (Fu et al., 2015). In late fusion, different modalities are merged in the decision level using various rules (e.g., majority voting, averaging) (Baltrušaitis et al., 2018). Here, the average of all modality outputs is calculated and used as the final output.

Before performing either feature concatenation or GMU based fusion, information from each modality is represented with a feature vector extracted from pretrained models, acting as modality streams. Then, we transform the vectors from all modalities into a single vector using the GMU or concatenation module. Finally, we pass the fused representation to a fully connected layer, creating a vector of size two (we have two classes). The sigmoid function is then applied to the two-dimensional vector to assign a label to each trailer. For late fusion, we capture the output of each single modality model before the sigmoid function (vectors of size two) and compute the average. Lastly, we pass the single representation to a sigmoid function for the classification.

5 Experiments

The goal of this section is to demonstrate that a multi-modal approach is an effective way to solve the task. We, therefore, compare the prediction performance of single modality models against all multimodal variations of the system.

As mentioned in the dataset section, the MM-Trailer dataset is imbalanced. Thus, to obtain reliable results, 5 fold cross-validation was selected as an evaluation method. In each fold, we select 10%

of the train sent as the validation set to obtain the best model. It should be noted that the dataset was split using the stratified approach, so as to ensure that each set has the same proportion of examples from each class. The metric used to evaluate the performance is the weighted F1 score, averaged over all 5 folds for each experiment.

5.1 Baseline Methods

Most Frequent Baseline: The first baseline is a naive approach to show that the problem is not easy to solve. In this model, we assign the most frequent class to all the instances in the validation and test sets, and we measure the F1 score by considering the ground truth label.

Text Baseline - Traditional Machine Learning: For the text baseline model, we provide a traditional machine learning method with hand-crafted features. We extract unigram and bigram features from subtitles and apply term frequency-inverse document frequency (TF-IDF) as the weighting scheme. Then, the feature vectors are passed to an SVM model for classification. We chose an SVM model as it performed well on the similar task of violence detection (Martinez et al., 2019).

Text Baseline - BERT + Attention + NRC: A popular resource to extract the emotion in the text is the NRC emotion lexicon (Mohammad, 2011). This dictionary maps words to eight different emotions (anger, anticipation, joy, trust, disgust, sadness, surprise, and fear) and two sentiments (positive and negative). Using this dictionary, we compute the normalized count of words per emotion over the entire subtitle and create a vector of size 10 for each trailer. We use this vector as an alternative to DeepMoji vector in the model.

Text Baseline - DeepMoji + fully connected layer: To show how much emotion by itself can contribute to the prediction of rating, we only use the DeepMoji vector as the input and pass it to a fully connected layer and sigmoid classifier for the prediction.

Video Baseline: Our video baseline is based on the deep 3-dimensional convolutional network (3D CNN) architecture proposed by (Tran et al., 2015). The 3D-CNN architecture applies 3D convolution and 3D pooling operations on video volumes instead of images. Each video is sub-sampled to an 18 evenly distributed frames that are used as input to the model. The training was performed for 50 epochs, using a 0.5 dropout rate, with a learning rate of 10^{-5} and a batch size of eight samples.

Audio Baseline: CNNs have shown promising results for audio classification (Hershey et al., 2017). To this end, for each full video, the log-Mel spectrogram is extracted from the audio using the LibROSA python library (McFee et al., 2015) and then used as input to a CNN architecture. For the log-Mel spectrograms 128 Mel-spaced frequency bins were used, while for the CNN model for this baseline, Inception V3 was adopted. The CNN model was trained for 100 epochs using a batch size of 64 samples and a learning rate of 10^{-5}. An early stopping policy was used during training to avoid over-fitting.

6 Results

Table 2 summarizes the results of our experiments. To examine the contribution of each modality for the rating task, we report the results for all single modality models; Audio only Model (A-MFCC), Text only Model with DeepMoji (T-BAD), and Video only Model (V-CNN/LSTM). As expected, our experimental results confirm that by leveraging all modalities we achieve a better result. As noted in Table 2 the highest weighted F1 score, 86.06%, is achieved by the GMU Fusion variant of the MMTR model with all modalities. This result improves the weighted F1-score of the best single modality model (T-BAD) over 3 percentage points ($P < 0.05$ based on t-test).

We also report the result for different combinations of two modalities using all fusion methods to show the effect of engaging all modalities (T-BAD + A-MFCC, T-BAD + V-CNN/LSTM , A-MFCC +V-CNN/LSTM , T-BAD + A-MFCC + V-CNN/LSTM). Based on the results, the combination of two modalities works better than every single modality, yet not as good as the combination of all modalities.

When comparing the different fusion approaches, we can see that GMU fusion outperforms the concatenation fusion systems. We speculate that the gains from GMU come from the ability of the gated unit to dynamically adapt the contribution of each modality to the intermediate representation. Statistical significance testing using t-test, demonstrated a significant difference between GMU and feature concatenation fusion ($p-value < 0.05$). However, the test does not confirm a significant difference

	Model	Test-WF
Single Modality Baselines	Most Frequent Baseline	60.37
	Text Baseline - Traditional Machine Learning	75.02
	Text Baseline - BERT+ Attention (T-BA)	81.99
	Text Baseline - BERT+ Attention+ NRC	81.67
	Text Baseline - DeepMoji+FC	68.23
	Video Baseline	75.33
	Audio Baseline	72.62
Single Modality Models	Audio- MFCC (A-MFCC)	73.86
	Text- BERT+ Attention+ DeepMoji (T-BAD)	82.67*
	Video- CNN/LSTM (V-CNN/LSTM)	79.41
Late (Fusion using two modalities)	T-BAD + A-MFCC	82.41
	T-BAD + V-CNN/LSTM	84.12
	A-MFCC + V-CNN/LSTM	79.68
Concatenation (Fusion using two modalities)	T-BAD + A-MFCC	82.17
	T-BAD + V-CNN/LSTM	82.80
	A-MFCC + V-CNN/LSTM	78.70
GMU (Fusion using two modalities)	T-BAD + A-MFCC	83.37
	T-BAD + V-CNN/LSTM	83.34
	A-MFCC + V-CNN/LSTM	80.35
Fusion using all Modalities (MMTR)	Late (T-BAD + A-MFCC + V-CNN/LSTM)	85.60
	Mid (Concat) (T-BAD + A-MFCC + V-CNN/LSTM)	82.75
	Mid (GMU) T-BAD + A-MFCC + V-CNN/LSTM	**86.06***

Table 2: Evaluation of the different variants of the MMTR system and other baselines using the MM-Trailer dataset. WF stands for weighted F1 score and results are averaged over 5 folds. A '*' indicates that the difference between the two classifiers' performance is shown to be statistically significant.

between late fusion and GMU. Thus, we can claim that for the trailer age-suitability problem, late fusion can generalize as good as GMU fusion.

The results for T-BAD and T-BA indicate that DeepMoji is a relevant feature for the rating task, and it helps the model to better discriminate red-band trailers from green-band ones. However, the result of DeepMoji+FC shows that the DeepMoji model is not sufficient to solve the task.

To obtain a better understanding of fusion results, we also provide other evaluation metrics using the MMTR system variant with GMU Fusion in Table 3 (as GMU version is the winner approach based on the result table). Based on the detailed result, most of the incorrectly predicted instances are red-band trailers. The first potential explanation behind this observation is that there are fewer instances of red-band trailers in our training set compared to green-band. As a result, it is more difficult for the model to capture all patterns in this class. The second reason may be the diversity of video content in red-band trailers. Recall that this class covers any content that is not appropriate for children. It is thus reasonable to assume that this class is more heterogeneous than the green band class. We plan to explore the possibility of a fine-grained classification of objectionable content as the next steps in this work.

7 Discussion

To analyze the weaknesses and strengths of the MMTR system, we first investigate the incorrectly predicted cases using the most effective version of the system (GMU Fusion) on each fold of the data. By averaging results over all folds, about 35% of incorrectly predicted cases with the MMTR system are also incorrectly predicted by each and every modality independently, fusion is thus unlikely to help in this case. We found that in about 50% of the instances where two modalities predict the wrong rating, the MMTR system justifiably trusts the single modality that is correct. In about 93% of the cases where only one modality is wrong, the MMTR GMU Fusion variant predicts the correct label, relying on the other two modalities.

After averaging results among all folds, we no-

Model	precision	recall	F1-score
Green	87.4%	95.0 %	91.0 %
Red	83.6%	65.0 %	72.8 %
Macro avg	85.6%	79.8 %	82.0 %
Weighted avg	86.6 %	86.4 %	86.0 %

Table 3: Performance of the MMTR system using alternative metrics by performing 5 fold cross-validation evaluation method. The results are averaged over 5 folds.

tice that the MMTR GMU Fusion variant system is not able to predict about 38 out of 294 instances per fold. We watched 40 incorrectly classified trailers (selected across all folds) to analyze why the model is not able to successfully predict the label. We introduce the following hypothesis for each of the individual modalities:

1) *Text Modality:* One main source of errors in text modality comes from the output of the speech recognition tool. First, the free version of the tool only works on short audio files. As a result, we split the whole audio to 10-second chunks. Thus, it is possible that we miss some words if the audio is cut off in the middle of the word. Also, low-quality audio impacts the word recognition rate of the automatic speech recognition system, which in turn cause the model not to recognize the specific bad words present in the video or the other way around, generate bad words by mistake (detect "please" as "pussy"). However, in some cases, the trailer either has very little speech (less than 10 words) or there is really no sensitive content in the language used. Not surprisingly, the text modality cannot work properly. Finally, in some green band videos, we observed that the trailer subtitles have the words "gun" and "shot", thus they are predicted incorrectly by the text modality. It seems that the text model is biased against the occurrence of these words that are presumably strongly correlated with violent content.

2) *Video Modality:* One main reason that the video modality model misses the sensitive content may relate to the video sampling rate. The inappropriate/violent scenes in these trailers disappear fast, or they may appear with a low frequency. As a result, we may miss them during sampling the frames in our model. The second potential reason is the quality of the trailers. We recognized that some of the trailers are old or are available in small files, so the frames are blurry, and even in some cases, not very clear to the human viewer. Lastly, we found out, there are some green-band trailers that still include brief sensitive content like the depiction of guns and blood, and our video modality model predicts them as red-band. These instances are mostly the R-rated movies that are sanitized for the trailer. However, the theme of the movie reflects itself in some frames. We can conclude that sometimes a single rating is not sufficient for expressing the type of the content, and as future work, we can predict a list of sensitive material in the video instead of a single label.

3) *Audio Modality:* In some cases, the music of the trailer is not compatible with the content. For example, we encountered musical movies with a high level of violence, but with smooth jazz music. Thus, it is difficult for the audio modality to distinguish between appropriate and inappropriate content. Moreover, in audio modality (similarly to the video modality), we capture samples from the continuous stream. Hence, if the intense audio (such as a scream or a gunshot) happens in a short period, our model may miss that.

We also investigated the genre of incorrectly predicted trailers in one of the data folds. The interesting point is that, for incorrectly classified red-band trailers, 55% are categorized as "Thriller" or "Horror" movies and 30% as "Comedy" (based on IMDB metadata). We do not incorporate metadata into our model to make the model suitable for any kind of online content. This observation shows that the genre of the movie can be a potential feature for the model if we have metadata available.

8 Conclusion

In this paper, we present a deep learning system named MMTR for automating the task of movie trailer age-suitability rating. MMTR fuses information from the video, audio, and text modalities. We also introduce a new data set to support research in this area. This dataset contains movie trailer videos along with their rating and metadata. The results of comparing our model with strong baselines demonstrated that the task is not easy, and a complicated multi-modal systems (GMU and late fusion) can achieve performance gains compared to other baselines. Beyond the practical use of a binary classification system, we are interested to move to the more challenging task of detecting the type of objectionable content and introducing explainability elements within the MMTR System.

Acknowledgments

This work was partially funded by the National Science Foundation under award 2036368.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Joakim Andén and Stéphane Mallat. 2011. Multiscale scattering for audio classification. In *Proc. International Society for Music Information Retrieval Conference*, pages 657–662. Miami, FL.

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A. González. 2017. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on pattern analysis and machine intelligence*, 41(2):423–443.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4733.

Paola Cascante-Bonilla, Kalpathy Sitaraman, Mengjia Luo, and Vicente Ordonez. 2019. Moviescope: Large-scale analysis of movies using multiple modalities. *arXiv preprint arXiv:1908.03180*.

Justin H Chang and Brad J Bushman. 2019. Effect of exposure to gun violence in video games on children's dangerous behavior with real guns: a randomized clinical trial. *JAMA network open*, 2(5):e194319–e194319.

François Chollet et al. 2015. Keras. https://keras.io.

Mihai Gabriel Constantin, Liviu Daniel Stefan, Bogdan Ionescu, Claire-Hélene Demarty, Mats Sjoberg, Markus Schedl, and Guillaume Gravier. 2020. Affect in multimedia: Benchmarking violent scenes detection. *IEEE Transactions on Affective Computing*.

Steven Davis and Paul Mermelstein. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing*, 28(4):357–366.

Claire-Hélène Demarty, Bogdan Ionescu, Yu-Gang Jiang, Vu Lam Quang, Markus Schedl, and Cédric Penet. 2014. Benchmarking violent scenes detection in movies. In *2014 12th International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 1–6. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv:1708.00524*.

Zhikang Fu, Bing Li, Jun Li, and Shuhua Wei. 2015. Fast film genres classification combining poster and synopsis. In *Intelligence Science and Big Data Engineering. Image and Video Data Engineering*, pages 72–81, Cham. Springer International Publishing.

Jürgen T Geiger, Björn Schuller, and Gerhard Rigoll. 2013. Large-scale audio feature extraction and svm for acoustic scene classification. In *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE.

Rajat Hebbar, Krishna Somandepalli, and Shrikanth Narayanan. 2018. Improving gender identification in movie audio using cross-domain data. In *Interspeech*, pages 282–286.

Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. 2017. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE*

conference on Computer Vision and Pattern Recognition, pages 1725–1732.

Khaled Koutini, Hamid Eghbal-Zadeh, Matthias Dorfer, and Gerhard Widmer. 2019. The receptive field as a regularizer in deep convolutional neural networks for acoustic scene classification. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Victor R Martinez, Krishna Somandepalli, Karan Singla, Anil Ramakrishna, Yalda T Uhls, and Shrikanth Narayanan. 2019. Violence rating prediction from movie scripts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 671–678.

Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8.

Saif Mohammad. 2011. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 105–114. Association for Computational Linguistics.

Inc Motion Picture Association. Speech recognition (version 3.8). https://www.filmratings.com/Content/Downloads/advertising_handbook.pdf. Accessed: 2020-10-07.

Michalis Papakostas, Evaggelos Spyrou, Theodoros Giannakopoulos, Giorgos Siantikos, Dimitrios Sgouropoulos, Phivos Mylonas, and Fillia Makedon. 2017. Deep visual attributes vs. hand-crafted audio features on multidomain speech emotion recognition. *Computation*, 5(2):26.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Zeeshan Rasheed and Mubarak Shah. 2002. Movie genre classification by exploiting audio-visual features of previews. In *Object recognition supported by user interaction for service robots*, volume 2, pages 1086–1089. IEEE.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

Niloofar Safi Samghabadi, Afsheen Hatami, Mahsa Shafaei, Sudipta Kar, and Thamar Solorio. 2019. Attending the emotions to detect online abusive language. *arXiv preprint arXiv:1909.03100*.

Mahsa Shafaei, Niloofar Safi Samghabadi, Sudipta Kar, and Thamar Solorio. 2019. Rating for parents: Predicting children suitability rating for movies based on language of the movies. *arXiv preprint arXiv:1908.07819*.

Victor C Strasburger. 1989. Adolescent sexuality and the media. *Pediatric Clinics of North America*, 36(3):747–773.

Vibha Tiwari. 2010. Mfcc and its applications in speaker recognition. *International journal on emerging technologies*, 1(1):19–22.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, page 4489–4497, USA. IEEE Computer Society.

Barbara J Wilson. 2008. Media and children's aggression, fear, and altruism. *The future of children*, pages 87–118.

Thomas Wolf, L Debut, V Sanh, J Chaumond, C Delangue, A Moi, P Cistac, T Rault, R Louf, M Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv, abs/1910.03771*.

Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702.

Anthony Zhang. Speech recognition (version 3.8). https://github.com/Uberi/speech_recognition. Accessed: 2020-04-30.

A Appendices

A.1 Implementation Details

This section discusses how the different streams of the MMTR system were implemented. For all experiments, we used the ADAM optimizer and the Cross-Entropy Loss function. Hyperparameter values were selected using manual tuning of the model using the best value of the weighted F1 Score for the validation partition, as the criterion.

Text Stream: For the BERT model we used the implementation provided by (Wolf et al., 2019). The LSTM layer consisted of 256 RNN units. Training was performed for 50 epochs, using a 0.3 dropout rate, with a learning rate of 10^{-5} and a batch size of eight samples.

Video Stream: For each movie trailer, frames were extracted with a rate of one frame per second, from which 18 evenly distributed frames were used to represent each video within the model. For the CNN feature extractor, we used the Inception V3 architecture pre-trained with ImageNet (Russakovsky et al., 2015). The model was trained using a learning rate of 10^{-5} and a batch size of 64 samples and by using an early stopping policy to avoid overfitting.

Audio Stream: For each trailer, the audio was split in 20 chunks. For the LSTM layer 256 RNN units were used. Training was performed for 50 epochs, using a 0.1 dropout rate, with a learning rate of 10^{-5} and a batch size of eight samples.

System Specifications: All models were developed using the Tensorflow (Abadi et al., 2015), Keras (Chollet et al., 2015) and PyTorch (Paszke et al., 2019) libraries on a machine with Ubuntu 14.04 LTS as the operating system. The system had an Intel Core™ i7 CPU running at 2.67GHz with four cores and 8 GB RAM memory. The video card used was a GeForce GTX 1080 Ti.

A.2 The GMU model:

The original equation of the GMU is represented in Equation 2; where W_v, W_t, and W_z are learnable parameters, x_v and x_t are modality feature vectors and $[.,.]$ stands for concatenation.

$$\begin{aligned} h_v = tanh(W_v.x_v), h_t = tanh(W_t.x_t) \\ z = \sigma(W_z.[x_v, x_t]]) \\ h = z * h_v + (1 - z) * h_t \end{aligned} \tag{2}$$

Note that in the extension to more than two modalities, the model ends up having more parameters as the the gates are no longer tied. But as shown empirically, this does not seem to be a problem for the model.