Predicting the Genre and Rating of a Movie Based on its Synopsis

Varshit Battu, Vishal Batchu, Rama Rohit Reddy, Murali Krishna Reddy, Radhika Mamidi
International Institute of Information Technology Hyderabad
{battu.varshit, ramarohitreddy.g, murali.dakannagari}@research.iiit.ac.in
vishal.batchu@students.iiit.ac.in
radhika.mamidi@iiit.ac.in

Abstract

Movies are one of the most prominent means of entertainment. The widespread use of the Internet in recent times has led to large volumes of data related to movies being generated and shared online. People often prefer to express their views online in English as compared to other local languages. This leaves us with a very little amount of data in languages apart from English to work on. To overcome this, we created the Multi-Language Movie Review Dataset (MLMRD). The dataset consists of genre, rating, and synopsis of a movie across multiple languages, namely Hindi, Telugu, Tamil, Malayalam, Korean, French, and Japanese. The genre of a movie can be identified by its synopsis. Though the rating of a movie may depend on multiple factors like the performance of actors, screenplay, direction etc but in most of the cases, synopsis plays a crucial role in the movie rating. In this work, we provide various model architectures that can be used to predict the genre and the rating of a movie across various languages present in our dataset based on the synopsis.

1 Introduction

As the amount of data present online increases exponentially day by day, we have reached a point where a human cannot comprehend all of it in a meaningful manner due to its sheer size. This lead to work on automated recommender systems. The main issue with these kinds of methods is that not all the information is present online and all the information present need not be correct. Automated movie genre and rating prediction have a lot of applications. We can recommend same genre movies based on his previous watch history. Genre of a movie can be identified by its synopsis. Recommending a movie only based on its genre is not a good idea as the same genre can have both good and bad movies. So Recommending movies based on both genre and rating would result in a proper recommendation system. But the main problem here is that people do not often tend to rate the movie they watch, thus automated rating prediction would be of great help for recommendation systems. Though the rating of a movie depends on multiple factors like actors, screenplay, direction etc. but that information is very difficult to capture through available data. In most of the cases, synopsis of the movie plays a crucial impact on audience rating. In this paper, we propose multiple deep-learning based methods to predict the genre and rating of a movie based on its synopsis.

There is a very little amount of data in languages apart from English to work on. To overcome this, we created the Multi-Language Movie Review Dataset (MLMRD). The dataset consists of genre, rating, and synopsis of a movie across multiple languages, namely Hindi, Telugu, Tamil, Malayalam, Korean, French, and Japanese. Balance in the dataset is not that good because nowadays movies in specific languages tend to belong to only specific genres due to various reasons like movie collections, ease of making etc. For example, no documentary movies are present in Telugu as such movies make fewer collections at Tollywood box office.

	Class	Telugu	Hindi	Tamil	Malayalam	French	Japanese	Korean
Genre	Action	230	45	21	56	1,314	763	15
	Comedy	60	35	27	25	2,602	15	6
	Crime	8	10	15	0	0	0	0
	Drama	47	88	62	43	3,425	2,798	60
	Family	18	0	0	21	0	763	0
	Horror	20	0	17	9	208	278	0
	Romance	133	42	19	14	127	0	2
	Thriller	43	33	20	18	532	0	14
	Documentary	0	0	0	0	833	38	19
Rating	1	10	16	7	19	766	267	1
	2	140	41	83	31	1,928	2,107	5
	3	353	126	57	69	3,302	2,209	15
	4	54	66	34	58	2,449	21	40
	5	2	4	0	9	596	51	55

Table 1: Number of data-points per genre/rating for each language. Not all languages have data belonging to all classes, the classes not corresponding to a language are marked(in red) with zero entries.

2 Related Work

Work has been done in related areas in the past. Basu et al. (Basu et al., 1998) propose an inductive learning approach to predict user preferences. Huang et al. (Huang and Wang, 2012) propose a movie genre classification system using a meta-heuristic optimization algorithm called Self-Adaptive Harmony Search. Rasheed et al. (Rasheed and Shah, 2002) present a method to classify movies on the basis of audio-visual cues present in previews which contain important information about the movie. Zhou et al. (Zhou et al., 2010) present a method for movie genre categorization of movie trailers, based on scene categorization. Gabriel S. Simoes et al. (Simões et al., 2016) explored CNNs in the context of movie trailers genre classification. Firstly, a novel movie trailers dataset with more than 3500 trailers was publicly released. Secondly, a novel classification method was done which encapsulates a CNN architecture to perform movie trailer genre classification, namely CNN-MoTion. Chin-Chia Michael Yeh et al. (Yeh and Yang, 2012) concerns the development of a music codebook for summarizing local feature descriptors computed over time. With the new supervised dictionary learning algorithm and the optimal settings inferred from the performance study, they achieved the state-of-the-art accuracy of music genre classification. Aida Austin et al. (Austin et al., 2010) created a database of film scores from 98 movies containing instrumental (non-vocal) music from 25 romance, 25 drama, 23 horror, and 25 action movies. Both pair-wise genre classification and classification with all four genres was performed using support vector machines(SVM) in a ten-fold cross-validation test. Jnatas Wehrmann et al. (Wehrmann and Barros, 2017) talked about a novel deep neural architecture based on convolutional neural networks (ConvNets) for performing multi-label movie-trailer genre classification. It encapsulates an ultra-deep ConvNet with residual connections, and it makes use of a special convolutional layer to extract temporal information from image-based features prior to performing the mapping of movie trailers to genres. Yong-Bae Lee et al. (Lee and Myaeng, 2002) presented a method for automatic genre classification that is based on statistically selected features obtained from both subject-classified and genre-classified training data. Pouya Ghaemmaghami et al. (Ghaemmaghami et al., 2015) addressed the specific problem of genre classification of movie clips using magnetoencephalography (MEG) data. They used the correlation analysis to show that genre related information is present in the visual and temporal areas of the brain and how these genre related brain signals can be decoded to target genre classes using the brain decoding paradigm. Junyong You et al. (You et al., 2010) presented a semantic framework for weakly supervised video genre classification and event analysis jointly by using probabilistic models for MPEG video streams.

3 Dataset

In order to create the dataset, we mined data from seven different websites. We used the data available in navbharattimes.indiatimes.com [1] for Hindi, telugu.samayam.com [2] for Telugu, tamil.samayam.com [3] for Tamil, malayalam.samayam.com [4] for Malayalam, m.movie.naver.com [5] for Korean, tsutaya.tsite.jp [6] for Japanese and www.allocine.fr [7] for French. We scraped the rating, genre, and synopsis of every movie from each website. Due to lack of resources and not much data is available in the specific language script, we could only mine a small amount of data. There aren't a lot of regional sites available that have trustworthy information to collect data from. Hence a lot of the languages have only a small number of data points in our dataset. However, we have ensured that the data collected, although small, is valid and collected from reputed movie review sites. We believe that having a small but strong and correct dataset is better than having a large dataset with a lot of noise and hence did not include other sites that did not have much reputation. The anonymized code can be found at https://goo.gl/nbWD9s and the dataset can be found at https://goo.gl/xpFv9q.

3.1 Data Extraction

The websites mentioned above have links to the synopsis of each movie along with the genre and rating in that web page. We first saved those links and then used beautiful soup to scrape the web page and get the synopsis, genre, and rating of the movie.

[1] http://navbharattimes.indiatimes.com/movie-masti/movie-review/articlelist/2325387.cms

[2] http://telugu.samayam.com/telugu-movies/movie-review/articlelist/48225171.cms?curpg=1

[3] http://tamil.samayam.com/tamil-cinema/movie-review/articlelist/48225229.cms

[4] http://malayalam.samayam.com/malayalam-cinema/movie-review/articlelist/48225004.cms

[5] http://m.movie.naver.com/m/category/movie/CurrentMovie.nhn?&page=1

[6] http://tsutaya.tsite.jp/search/?dm=0&st=0&p=1&ic=1

[7] http://www.allocine.fr/critique/fichepresse_gen_cpresse=82049.html

3.2 Preprocessing

After collecting all the required data, we had to preprocess the data to cluster genres into classes. Since the data collected had different classes in each language we merged similar classes into one broader class as explained in Section 3.2.1. Finally, we were left with 9 classes viz. action, comedy, crime, drama, family, horror, romance, thriller and documentary for each language. The details are mentioned in Table 1. We only added movies having all the three - genre, rating and synopsis into the dataset and ignored movies which were missing information. To validate the data, we performed a manual inspection at various data points selected at random to ensure the ratings and genres are valid and not erroneous. Here a synopsis, its respective rating and it's genre is referred as a data point. Once this was done we shuffled the data before passing it through the model. Shuffling the data helps make the training and test sets more representative of the overall distribution of the dataset. We then split the data into two parts containing 80% and 20% of the entire data for training and testing respectively.

3.2.1 Grouping of genres

The original data had several kinds of genres. We grouped all relevant genres together to finally end up with 9 different classes of genres as mentioned in Table 2. For example Autobiography, Biopic etc were put into the Documentary class. Romantic-Comedy as the name suggests could be part of the Romance group but when we manually inspected a few data points at random they were more suited for the comedy genre and hence put it there.

3.3 Statistics

There are 14,991 entries in the dataset we compiled. The language-wise distribution of entries and words per language are mentioned in Table 3. This is a big dataset covering a total of seven languages belonging to different language families. Each language has different average lengths of the synopsis in terms of the number of words. For example Hindi has 253 entries and 1,67,842 words whereas Tamil has 181 entries and 1,01,904 words.

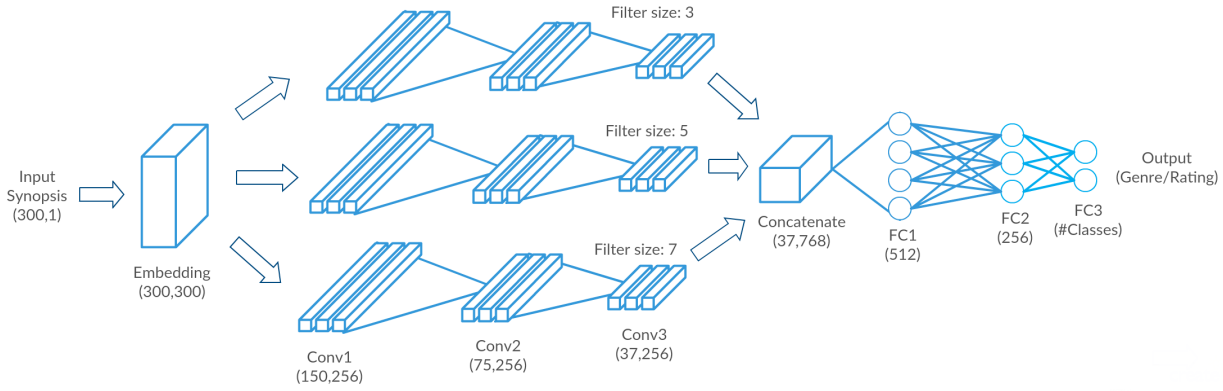

Figure 1: Branched CNN based genre/rating prediction model architecture with character inputs which considers the synopsis to perform the prediction. The Conv blocks represented in the figure consist of Convolution, ReLU and Max-pool layers. We used Dropouts at various places for regularization. Word inputs are similar except that the input size is different.

Genre Class	Genres
Action	Action, Adventure, Sci-Fi Superhero, Sport, War
Comedy	Comedy, Romantic-Comedy
Crime	Crime
Drama	Drama, Fantasy Music-Drama, Action-Drama
Family	Family, Animation Musical, Anime, Kids
Horror	Horror
Romance	Romance, Music-Romance
Thriller	Thriller, Mystery
Documentary	Documentary, Autobiography History, Biopic, Biography

Table 2: Preprocessing the genres to form 9 genre classes that are used for genre prediction.

Language	Entries	Words	Avg(Words)
Telugu	559	1,00,431	179
Hindi	253	1,67,842	663
Tamil	181	1,01,904	563
Malayalam	186	45,553	244
French	9,041	2,64,801	29
Japanese	4,655	7,68,437	165
Korean	116	9,069	78
Total	14,991	14,58,037	-

Table 3: Statistics of MLMRD. The entries and words are per language, Avg(Words) is the average length of each synopsis in terms of the number of words.

4 Genre and Rating Prediction

We predict the genre and rating of a movie based on its synopsis alone. Genre prediction deals with 9 output classes as shown in Table 2. We treat rating prediction as a classification task rather than regression. We round the ratings leaving us with 5 classes that we try to predict.

4.1 Character Embeddings

Each character of the input synopsis is converted to a vector dynamically using an Embedding layer at the inputs to the networks. These character vectors are then passed along to various convolution and recurrent networks. Using a one-hot encoded representation of the characters also gave similar accuracies.

Convolution networks: The input to the CNN (LeCun et al., 2015) consists of all the character embeddings stacked as filters which are then passed along the network to predict an output genre/rating class. The network as mentioned in Figure 1 has a branched structure where filters of various sizes are used in the convolution layers in each of the branches and the outputs are concatenated before being passed onto fully connected layers to predict the output genre/rating class.

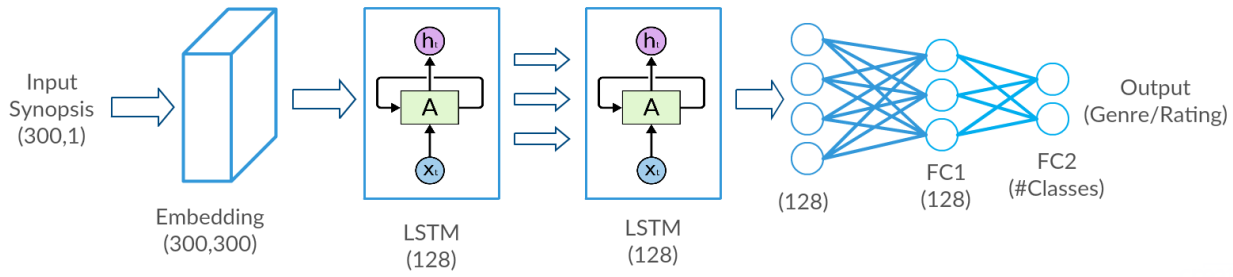

Figure 2: LSTM based genre/rating prediction model architecture with character inputs which considers the synopsis to perform the predictions. Replacing LSTM cells with GRUs would give us GRU models

Recurrent networks: For LSTM (Hochreiter and Schmidhuber, 1997), GRU (Chung et al., 2014) and RNN (Quast, 2016) based networks as shown in Figure 2, we feed in character vectors one at a time as input and the predicted output is passed forward to multiple fully connected dense layers which predict the output genre/rating class.

4.2 Word Embeddings

Each word in the input is converted into a vector. These vectors are generated either dynamically using an Embedding layer or statically using Gensim (Rehurek and Sojka, 2010). These generated vectors are used as inputs to convolution and recurrent networks similar to how character encodings were used.

4.3 Sentence Embeddings

Sentence vectors were generated using Doc2Vec (Le and Mikolov, 2014). Doc2Vec takes all the sentences at once and generates sentence vectors for them. However, this requires all the data to be fed into Doc2Vec i.e both train and test sentences and hence this cannot be performed on unseen data.

Fully connected networks: Since the entire synopsis is encoded using a single vector, we pass the vector through a fully connected network which predicts the output genre/rating class and convolution/recurrent networks provide no benefits here.

4.4 Concatenated Embeddings

We observed that different types of embeddings performed well for different languages, for example, word embeddings for Telugu and Hindi, sentence embeddings for Tamil etc. Hence, concatenating all the three embeddings namely character, word and sentence embeddings and pass them through different models so that there can be an increase in the accuracy as the network chooses important parts of these embeddings.

5 Experiments

We performed numerous experiments using various deep learning models including convolution and recurrent based networks with character, word, and sentence level embeddings for inputs. We also compare our proposed models with some of the popular traditional approaches such as SVMs(Cortes and Vapnik, 1995) and Random Forests(Svetnik et al., 2003) and show that deep learning based methods beat them by large margins as shown in Tables 4 and 5.

5.1 Experimental Details

We use Keras with the Tensorflow backend to perform all our experiments. We use a GeForce GTX-1080Ti GPU in order to train our models (Each model takes less than 15 minutes to complete training). We use dropouts at various locations in the networks to reduce over-fitting. Categorical cross entropy loss is used as the loss function along with the Adam optimizer for training all the networks. We observe that dynamic embeddings perform better than static embeddings in all word based models and hence we use embedding layers in all the models instead of using Gensim or GloVe word vectors. ReLU activations are used throughout the networks except for the last layers which use SoftMax activations in all the models. The code provided along with the paper has further implementation details.

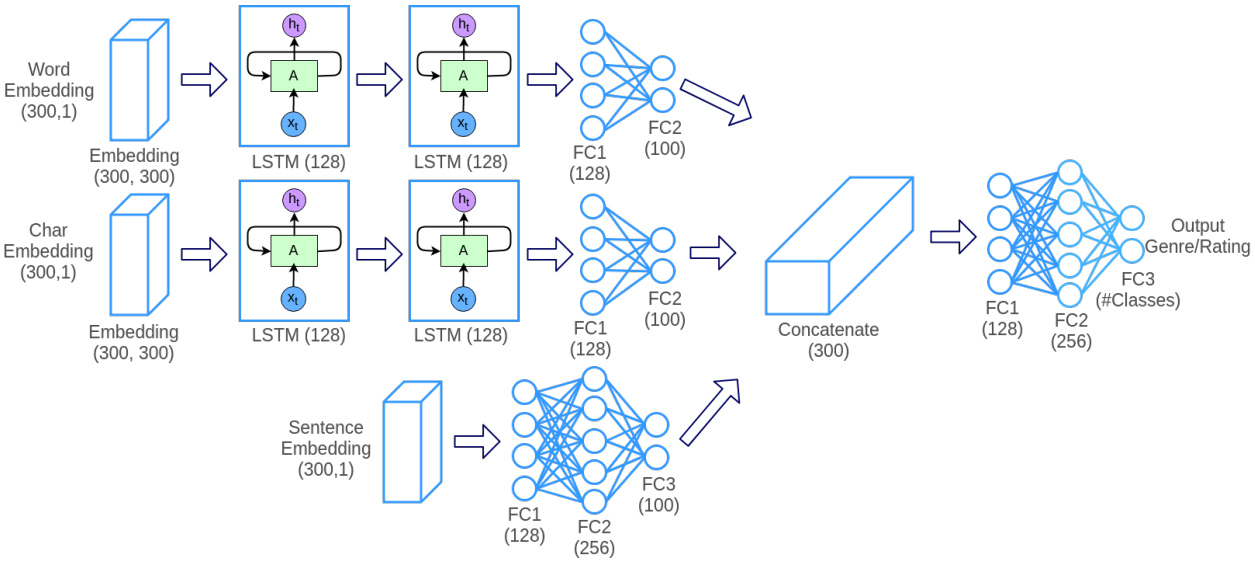

Figure 3: Hybrid model based on stacking three models to predict genre/rating with word, character and sentence embeddings as the input considers synopsis to perform the prediction. We used Dropouts at various places for regularization.

5.1.1 SVM

SVMs are commonly used for recognition and regression analysis of data. Considering features from the reviews as inputs, they try to classify them into one of the genre and rating classes. We run a trained Doc2Vec model using the entire review as an input and that provides us a 300-dimensional embedding that we use as an input to the SVM.

5.1.2 Random Forests

Similar to the features we use for the SVM, we use the Doc2Vec embeddings of reviews as inputs for the Random Forest classifier that predicts the genre and rating classes.

5.1.3 CNN

We use dropouts of 0.5, 0.7, and 0.8 at three places in the network to ensure that the model does not overfit.

Character embedding based model: Inputs are padded to a length of 300 characters and trimmed if they exceed this length. Character embeddings are generated using an embedding layer that generates 300-dimensional embeddings for each character. We train the model for 300 epochs with a batch size of 512.

Word embedding based model: Inputs are padded to a length of 150 words and trimmed if they exceed this length. Word embeddings are generated using an embedding layer that generates 300-dimensional embeddings for each word. We train the model for 200 epochs with a batch size of 512.

5.1.4 LSTM

Character-based inputs are padded to a length of 300 and word-based inputs are padded to a length of 100. The embedding layer generates 300-dimensional embeddings. The network consists of two LSTM layers followed by multiple Dense layers. A recurrent dropout of 0.4 is used in the first LSTM layer. We train the models for 300 epochs with a batch size of 512.

5.1.5 GRU

The parameters used are identical to the LSTM parameters except that both the LSTM layers are replaced with GRU layers.

5.1.6 FCNN

The model receives a single 300-dimensional sentence embedding that was generated using Doc2Vec. This is passed through a few Dense layers to get our final output. We use dropouts of 0.4 at various places in the network. We train the models for 200 epochs with a batch size of 512.

5.1.7 Hybrid Model

The model as shown in Figure 3 receives three inputs i.e word embedding, char embedding and sentence embedding. The word and char embeddings go into two different LSTM networks. The sentence embeddings go into a fully connected dense network. Each model produces a 100-dimensional output which are concatenated. This 300-dimensional concatenated embedding is given as an input to a dense network to predict the genre and ratings. We use a Dropout of 0.4 throughout the model and train the model for 300 epochs with a batch size of 512.

5.2 Results

Promising results were obtained in both genre and rating prediction using just the synopsis as the input, as presented in Tables 4 and 5. For instance, we obtain 91.2% and 90.2% while predicting the Genre and Rating in Telugu respectively.

5.3 Analysis

(Dryer, 1997) classified languages in 6 ways depending on whether a subject follows a verb or whether an object follows a verb. The languages we worked on come under SV/OV(Subject-Object-Verb) type of languages. Our dataset consists of multiple languages, some of which are agglutinative (Telugu, Malayalam, Tamil, Japanese and Korean). Our methods obtain good results with various types of languages.

Character vs Word Embeddings: We observe that on the whole, word embeddings perform better in general, however in certain cases considering agglutinative languages such as genre prediction in Japanese and rating prediction in Malayalam perform better with character embeddings.

Sentence FCNNs: Datasets having small amounts of data work well with sentence vectors. Larger datasets, however, pose issues since the embeddings generated are not precise enough in these cases to differentiate the inputs well.

Hybrid Model: LSTM networks learn sequence-based information very well from the character and word embeddings whereas the FCNN learns well from the sentence vectors. Our intuition was that if we develop a model which would use these three models collectively to predict the genre and rating there would be a significant increase in accuracy. So we developed a stacked model which uses combined information from two LSTM networks and one FCNN network to predict the genre and rating. Stacking is an ensemble learning technique and is also known as meta ensembling. This new model outperforms the earlier models as it gives more weight to the individual model where it performs well and gives lesser weight to the individual model where it performs badly. The reason we cannot see a huge change in the rating prediction unlike genre prediction is that the final objective function is not able to learn well from three different models due to the difference in the flow of gradients. If the training is done separately, we would see an increase in accuracy.

Traditional vs Deep Learning approaches: There is a huge difference in the performance of traditional machine learning approaches such as SVMs and Random Forests when compared to deep learning based methods such as Sent-FCNN. They all use the same inputs which are the embeddings generated using the Doc2Vec model. We believe that one of the main reasons for this is that deep learning based approaches tend to generalize a lot better as compared to traditional methods and hence they perform a lot better on unseen test data. We have also tried to see how these methods perform if the testing data is just a subset of training data. In this case, we notice that they are both able to represent their training data well and hence achieve similar accuracies during testing. However, this only happens since the number of data points in our dataset are not a lot. When the number of data points increases, deep learning based approaches show tremendous amounts of generalizability which allows them to attain much higher accuracies compared to traditional methods. We validate this by adding noise to our inputs by randomly scaling the inputs up to 15%. This resulted in the failure of traditional approaches as we discussed earlier.

Inter-language comparisons: Having a dataset that consists of multiple languages allows us to verify how well our approaches scale and how they can be generalized and applied to data from various domains. MLMRD would also be useful to other researchers who would want to test out their approaches on different languages since multilingual data is becoming popular in recent times. We ob-

Model	Train %	Telugu	Hindi	Tamil	Malayalam	French	Japanese	Korean
Char-CNN	80	87.7	87.5	89.2	87.8	89.0	93.3	89.8
Word-CNN		88.9	87.9	87.6	88.3	89.0	91.0	90.7
Concat-CNN		89.7	89.3	89.5	88.9	**89.4**	92.4	89.8
Char-LSTM		87.7	88.0	87.9	88.1	89.1	**93.8**	91.7
Word-LSTM		89.1	88.7	87.8	88.2	88.9	91.6	91.2
Concat-LSTM		89.1	88.8	89.8	89.7	88.8	92.4	90.7
Char-GRU		87.4	87.5	87.4	87.6	88.9	93.5	91.7
Word-GRU		87.3	87.8	87.6	87.3	89.0	90.8	90.7
Concat-GRU		88.9	88.9	89.5	89.5	88.9	92.0	88.9
Sent-FCNN		87.6	88.1	89.3	87.0	89.4	92.8	**92.6**
Concat-FCNN		83.9	84.3	89.8	82.5	84.3	92.1	85.0
Hybrid-model		**91.2**	**89.9**	**90.0**	**89.8**	**89.8**	91.8	92.1
SVM	80	41.9	45.1	48.6	39.4	49.0	63.2	62.5
Random Forests		53.5	50.9	40.5	42.1	40.5	67	58.3

Table 4: Genre prediction accuracies for various models on each of the languages in MLMRD.

Model	Train %	Telugu	Hindi	Tamil	Malayalam	French	Japanese	Korean
Char-CNN	80	85.2	83.1	82.7	82.1	89.1	90.5	90.4
Word-CNN		82.2	80.1	83.2	81.1	89.0	89.4	89.8
Concat-CNN		**90.2**	**89.5**	**89.2**	**89.5**	**89.2**	**91.5**	89.8
Char-LSTM		85.9	81.7	82.2	82.1	89.1	89.1	**93.5**
Word-LSTM		86.2	83.5	80.4	83.2	88.9	89.0	91.4
Concat-LSTM		89.0	88.9	88.9	89.2	88.9	91.2	**93.5**
Char-GRU		85.8	82.4	81.2	81.1	89.2	89.5	91.2
Word-GRU		86.1	83.2	81.1	80.5	89.0	89.1	91.7
Concat-GRU		88.9	**89.5**	88.9	89.5	88.9	91.2	90.7
Sent-FCNN		85.2	80.9	84.2	80.2	**89.2**	89.5	91.3
Concat-FCNN		80.0	78.0	89.1	69.5	73.1	77.9	73.3
Hybrid-model		84.6	83.1	81.0	80.0	80.5	80.8	82.5
SVM	80	58.0	50.9	45.9	47.3	44.7	48.8	45.8
Random Forests		68.8	52.9	54.0	44.8	43.1	55.5	54.0

Table 5: Rating prediction accuracies for various models on each of the languages in MLMRD.

serve that Telugu, French, Japanese and Korean perform much better than Hindi, Tamil, and Malayalam in rating prediction. This dataset would also allow us to work towards generalized methods that work on multiple forms of inputs that don't require different models to handle different languages which is how traditional approaches work.

Qualitative examples with translations and analysis are shown in Appendix A.

6 Conclusion

We provide the multi-lingual dataset MLMRD, consisting of movie genres, ratings and the synopsis which can be used to test various machine learning and NLP based techniques on different kinds of data. We believe that this would be a valuable asset since a lot of these languages are low-resource languages with almost no data available to experiment on. We also propose multiple methods to establish baselines for movie genre and rating prediction based on the synopsis. Additionally, we show how our proposed methods are generalizable and work well on different kinds of data. We plan to extend the approach using movie plots as inputs which would provide us with important information. We also plan to normalize the data collected so that each of the classes have a similar number of data-points.

References

[Austin et al.2010] Aida Austin, Elliot Moore, Udit Gupta, and Parag Chordia. 2010. Characterization of movie genre based on music score. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 421–424. IEEE.

[Basu et al.1998] Chumki Basu, Haym Hirsh, William

Cohen, et al. 1998. Recommendation as classification: Using social and content-based information in recommendation. In *Aaai/iaai*, pages 714–720.

[Chung et al.2014] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*.

[Cortes and Vapnik1995] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

[Dryer1997] Matthew S Dryer. 1997. On the six-way word order typology. *Studies in Language. International Journal sponsored by the Foundation "Foundations of Language"*, 21(1):69–103.

[Ghaemmaghami et al.2015] Pouya Ghaemmaghami, Mojtaba Khomami Abadi, Seyed Mostafa Kia, Paolo Avesani, and Nicu Sebe. 2015. Movie genre classification by exploiting meg brain signals. In *International Conference on Image Analysis and Processing*, pages 683–693. Springer.

[Hochreiter and Schmidhuber1997] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, pages 1735–1780.

[Huang and Wang2012] Yin-Fu Huang and Shih-Hao Wang. 2012. Movie genre classification using svm with audio and video features. In *Proceedings of the 8th International Conference on Active Media Technology*, pages 1–10.

[Le and Mikolov2014] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196.

[LeCun et al.2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*, pages 436 EP –.

[Lee and Myaeng2002] Yong-Bae Lee and Sung Hyon Myaeng. 2002. Text genre classification with genre-revealing and subject-revealing features. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 145–150. ACM.

[Quast2016] Bastiaan Quast. 2016. rnn: a recurrent neural network in r. *Working Papers*.

[Rasheed and Shah2002] Zeeshan Rasheed and Mubarak Shah. 2002. Movie genre classification by exploiting audio-visual features of previews. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 1086–1089.

[Rehurek and Sojka2010] Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS*, pages 45–50.

[Simões et al.2016] Gabriel S Simões, Jonatas Wehrmann, Rodrigo C Barros, and Duncan D Ruiz. 2016. Movie genre classification with convolutional neural networks. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 259–266. IEEE.

[Svetnik et al.2003] Vladimir Svetnik, Andy Liaw, Christopher Tong, J Christopher Culberson, Robert P Sheridan, and Bradley P Feuston. 2003. Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences*, 43(6):1947–1958.

[Wehrmann and Barros2017] Jônatas Wehrmann and Rodrigo C Barros. 2017. Movie genre classification: A multi-label approach based on convolutions through time. *Applied Soft Computing*, 61:973–982.

[Yeh and Yang2012] Chin-Chia Michael Yeh and Yi-Hsuan Yang. 2012. Supervised dictionary learning for music genre classification. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, page 55. ACM.

[You et al.2010] Junyong You, Guizhong Liu, and Andrew Perkis. 2010. A semantic framework for video genre classification and event analysis. *Signal Processing: Image Communication*, 25(4):287–302.

[Zhou et al.2010] Howard Zhou, Tucker Hermans, Asmita V Karandikar, and James M Rehg. 2010. Movie genre classification via scene categorization. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 747–750.

A Appendix

A.1 Qualitative example - correct prediction

A.1.1 Hindi

Original Synopsis -

फ्लाइट अटेंडेंट नीरजा भनोट की सच्ची कहानी पर बनने वाली इस फिल्म को बनाने की प्लानिंग करीब दस साल पहले हुई। कहानी पूरी होने के बावजूद किसी न किसी वजह से फिल्म की शूटिंग टलती रही। नीरजा की रियल लाइफ स्टोरी की रील लाइफ में दर्शाया गया है। चंडीगढ़ की रहने वाली नीरजा ने 5 सितंबर 1986 को हाइजैक हुई पैन एम फ्लाइट 73 में सवार 359 यात्रियों को अपनी सूझबूझ और बहादुरी के दम पर बचाया था, लेकिन खुद शहीद हो गईं थीं। ऐसा पहली बार हुआ जब भारत सरकार ने सिर्फ 23 साल की उम्र में किसी को अशोक चक्र दिया हो, लेकिन नीरजा भनोट को यह सम्मान मरणोपरांत दिया गया। आज इस फिल्म की कहानी या कुछ सीन्स को पाक विरोधी करार देने के बाद पाकिस्तान में फिल्म को इस शुक्रवार रिलीज नहीं किया गया, लेकिन पाकिस्तान सरकार ने नीरजा को तमगा-ए-इन्सानियत अवॉर्ड से सम्मानित किया था। यह फिल्म नीरजा भनोट (सोनम कपूर) के आस-पास घूमती है। नीरजा के साथ-साथ उसकी इस कहानी में मां रमा भनोट (शबाना आजमी) और पापा (योगेंद्र टिक्कू) भी हैं जो हर कदम पर अपनी बेटी के साथ हैं। स्टडी के बाद मॉडलिंग और नीरजा की पर्सनल लाइफ को भी इस कहानी का हिस्सा बनाया गया है।

Translated Synopsis - Plan to make the film on the true story of Flight Attendant Neerja Bhanot was made almost ten years ago. Even after the story was complete, the shooting of the movie was being inhibited due to some reason. Neeraja's Real Life Story is featured in reel Life. Neerja, a resident of Chandigarh, saved 359 passengers on Pan Am Flight 73 in Hijack on September 5, 1986, on the basis of her sense of bravery and she herself became a martyr. This is the first time that the Indian government has given Ashok Chakra only at the age of 23, but Neerja Bhanot was honored. Today, the film was not released on this Friday in Pakistan after the story of the film or some scenes were labeled anti-Pakistan, but the Pakistan government had awarded Neerja the Tamgha-e-Insignias Award. This movie revolves around Neeraja Bhanot (Sonam Kapoor). Along with Neeraja, in this story, there are also mom Rama Bhanot (Shabana Azmi) and Papa (Yogendra Tiku) who are with their daughter at every step. Modeling after study and Neeraja's personal life has also been made part of this story.
Actual Genre - Drama
Predicted Genre - Drama
Actual Rating - 3
Predicted Rating - 3

A.2 Qualitative example - wrong prediction

A.2.1 Telugu

Original Synopsis -

క్లుప్తంగా చెప్పాలంటే.. ప్రతీ వ్యక్తి తన జీవితాన్ని తక్కువ చేసుకుంటూ ఊహల్లో తనది కాని మరో ప్రపంచంలో బతికేస్తుంటాడు. అలాంటి జీవితాలను వెళ్ళదీస్తున్న ఇద్దరి కథల్లో ఎలాంటి మార్పులు వచ్చాయన్నదే ొనాలో ఒకడుొ కథ. విక్కీ అలియాస్ విజ్ఞేష్ (సిద్ధార్థ్) తన చిన్న థియేటర్‌లో టార్చ్‌లెట్ బాయ్‌గా పనిచేస్తుంటాడు. అంద వికారంగా ఉన్నానని, జీవితాన్ని జాలీగా గడపడానికి డబ్బులు లేవని బాధ పడుతూ కష్టపడి పనిచేసుకునే మంచి వ్యక్తి. సాధారణంగా సాగిపోయే అతడి జీవితంలో లూసియా పేరుగల డ్రగ్ విపరీతమైన మార్పులు తీసుకొస్తుంది. కాప్సిల్ తీసుకున్న తర్వాత అతడు తాను కోరుకునే జీవితాన్ని కలలో బతికేయొచ్చు. కలలో విక్కీ స్టార్ హీరో. సమాజంలో ఎంతో డబ్బు, పేరు. అలాంటి స్టార్ జీవితాన్ని కలలో గడిపేస్తుంటాడు. రెండు కథలు సమాంతరంగా సాగుతూండే క్రమంలో ఇటు థియేటర్‌లోని విక్కీ జీవితంతో పాటు, అటు స్టార్ హీరో జీవితం గడుపుతూ ఎలాంటి మార్పులు వచ్చాయో అదే కథ.

Translated Synopsis - In short, every person thinks low about their lives and live in an other imaginary world. 'Nalo Okadu' is one such story about the life of two people and how it happened to change. Vicky alias Vignesh (Siddharth) works as a torchlight boy in his small theater. He is a good hard working poor guy who feels bad about his looks and status of life. A drug named Lucia brings tremendous changes in his smooth going life. After taking the drug he can live a life that he wants in his dreams. Vicky is a star hero in his dream. He lives a life of star who has a lot of money and fame in the community. The movie is about the parallel lives of Vicky, the one in the theater and the star life and how it changes his life.
Actual Genre - Romance
Predicted Genre - Comedy
Actual Rating - 2
Predicted Rating - 3

Analysis - Main reason for wrong prediction in above case is that the synopsis does not have many details about the genre. As we can see, it is quite ambiguous and is difficult to predict the genre from it. These types of synopsis result in bad prediction.

32nd Pacific Asia Conference on Language, Information and Computation
Hong Kong, 1-3 December 2018