THE ENDLESS MINORITY BATTLE FOR DIVERSITY IN HOLLYWOOD: ANALYSIS THROUGH SPEECHES

Oumaima Merhben, Mohamed Yassine Mejri, Myriem Azzouz, Walid Slim

University Paris Dauphine

1 Context

The controversial 92nd Academy Awards (2020) nominations have revived the old debate representation of minorities in Hollywood: No woman named for best production, one in twenty black actresses nominated for the interpretation awards. In this context we want to explore the differences between the dialogues written for female and male characters in Hollywood movies. We use movie conversations with memorable metadata and quotes to analyze how the dialogues change according to gender. We are looking to investigate whether dialogues of female characters are less memorable than those of characters male.

2 Data exploration

In this project we worked with two datasets :

2.1 Cornell Movie Dialogues Corpus

It has the following properties:

1. 220,579 conversational exchanges between 10,292 pairs of movie characters.
2. 9,035 characters from 617 movies.
3. 304,713 total utterances.
4. Movie metadata including: genres, release year, IMDB rating, IMDB votes.
5. Character metadata: gender, position on movie credits.

2.2 Cornell Memorability Dataset

It has the following properties:

1. 894014 movie script lines from 1068 movie scripts.

2. 6282 one-line memorable quotes that are automatically matched with the script line which contain them.

3. 2197one-sentence memorable quotes paired with surrounding non-memorable quotes from the same movie, spoken by the same character and containing the same number of words.

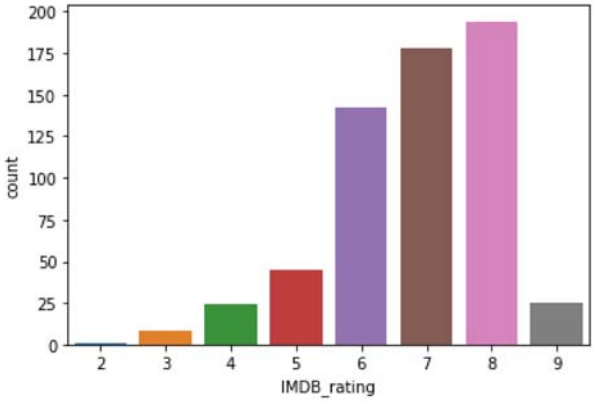

Figure 1: IMDB rating

3 Classification

In our first dataset, given a dialogue, we have the gender of the character, the rating and genre of the movie, etc.. We used this data to build classifiers to see if given a single dialogue in a movie, can the classifiers classify those respective characteristics. If our classifiers can learn representations of the data so that they can accurately understand the gender of a character(CGENRE) or the rating of a movie (CIMDB).

In the second dataset, we have 2,197 matching memorable-non-memorable quotes, We used this data to create a classifiers that can detect whether a movie quote will be memorable or not(CMEM).

For the Word Embedding phase which allows represent each word of a dictionary by a vector of numbers we used pre-trained GloVE vectors.

3.1 Glove

GloVe allows us to take a corpus of text, and intuitively transform each word in that corpus into a position in a high-dimensional space.

Global Vectors for words representation (GloVe) is provided by Stanford. They provided various models from 25, 50, 100, 200 to 300 dimensions based on 2, 6, 42, 840 billion tokens.

In our case we used “glove.6B.100d.txt“ and “glove.6B.300d.txt“ which contains respectively a 100-dimensional,300-dimensional version of the embedding :100,300 hidden units.

Our embedding layer takes as parameters :

-vocabulary size: which is the number of distinct tokens in our corpus

-output size: This is the size of the vector space in which words will be embedded : embedding space

-input size: the length of the input data

-weights : is an embedding matrix that represents the embedding of each word in the corpus.

Instead of doing the matrix multiplication between the inputs and hidden layer we directly grab the values from embedding weight matrix.

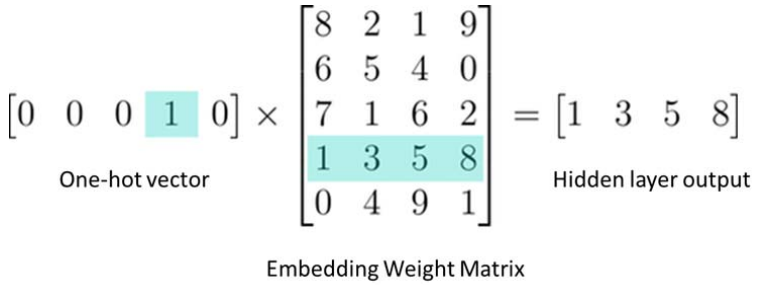

Figure 2: IMDB rating

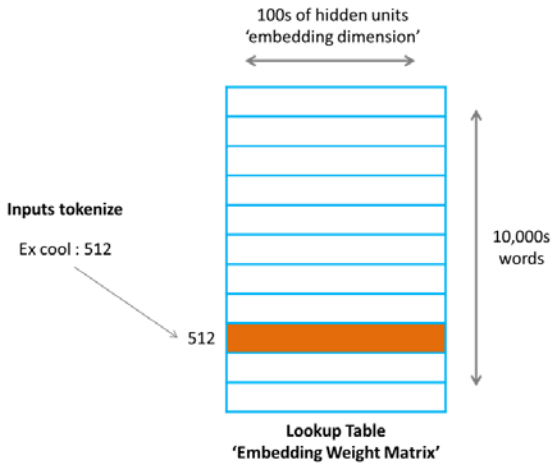

Figure 3: IMDB rating

The output of the embedding layer is then used as input to an LSTM and CNN models.

3.2 Classifier Models:

In order to build these classifiers, the idea is to compare for each of them three models:

3.2.1 Long Short-Term Memory (LSTM):

LSTM help capture long term dependencies in the data which is common in natural language. The input of the LSTM is a sequence of words in each dialogue. Each word was represented as a vector using a 100 dimensional pretrained GloVe vector. The representation was then fed into an LSTM cell along with the previous time step's hidden layer representation represented by the following equations:

$$
\begin{aligned}
i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{(t1)}) && \text{(Input gate)} \\
f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_{(t1)}) && \text{(Forget gate)} \\
o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{(t1)}) && \text{(Output/Exposure gate)} \\
\tilde{c_t} &= \sigma(W^{(c)}x_t + U^{(c)}h_{(t1)}) && \text{(New Memory Cell)} \\
c_t &= ftc_{(t1)} + i_t\tilde{c_t} && \text{(Final memory cell)} \\
h_t &= o_t tanh(c_t)
\end{aligned}
$$

Figure 4: LSTM

3.2.2 Convolutional NeuralNetwork (CNN):

In addition to the LSTM, we also used a Convolutional Neural Network to classify our dialogues. While CNNs have generally been used in computer vision, there have been successes using CNNs to model sentences especially for sentence classification tasks.

3.2.3 Support Vector Machines (SVM):

We used a linear classifier, specifically an SVM. As input, we converted each dialogue into a vector by averaging a 100 dimensional pretrained GloVe vector of each word.

4 CGENRE classifier

Gender of the character speaking the dialogue: binary classification using data from the first dataset. The dataset consisted of around 70000 female quotes and 170000 male quotes.

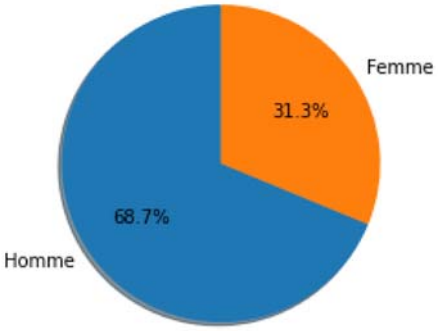

Figure 5: Gender in movies

The models were trained on 50000 male examples and 50000 female examples, 500 of each for validation set and 500 of each for test set resulting in a 100000 test set and 1000 examples each for validation and testing.

To preprocess the data we use tokenization . In tokenization we convert group of sentence into token.

Then we apply padding to create sentences of the same length (MAX SEQUENCE LENGTH) we complete the vectors with zeros. We define then our models .

The embedding layer is seeded with the GloVe word embedding weights. We chose the 300-dimensional version, therefore the Embedding layer must be defined with output dim set to 300.

After a certain number of time steps which was tuned as a hyperparameter, the last hidden layer representation in LSTM and CNN models was used as input to a sigmoid layer to output into the label space.

5 CIMBD classifier

5.1 Data Preparation

In order to predict the IMDB rating for each movie, we had grouped by movie id characters dialogue and used Glove pretrained weights.

5.1.1 Glove Embedding

Global Vectors for words representation (GloVe) is provided by Stanford. They provided various models from 25, 50, 100, 200 to 300 dimensions based on 2, 6, 42, 840 billion tokens. In our case we used “glove.6B.100d.txt“ which contains a 100-dimensional version of the embedding :100 hidden units.

5.2 Creating models

Our models input is a corpus : list of shape (617,) as we have 617 movies . Each line represents the content of the movie's dialogues.

The output of the embedding layer is then used as input to an LSTM and CNN models

The previous embedding layer outputs a 3-D tensor into the LSTM layer. LSTM generally have the problem of overfitting , so we added a SpatialDropout1D. We can also use the LSTM specific dropout that has a more pronounced effect on the convergence of the network than the layer-wise dropout. At the end we used a dense layer with a 'sigmoid' activation for classification (10 layers).

Convolutional neural networks excel at learning the spatial structure in input data.

While constructing this model we used Conv1D and GlobalMaxPooling . In fact the the Global Max Pooling layer reshapes the 3D tensor into a 2D one.

6 CMEM classifier

The goal of this part is to build a binary classifier in order to predict whether a dialogue is memorable or not.

The Dataset had 2197 pairs of memorable/non-memorable dialogues. It was then split into memorable and non-memorable dialogues and were labeled according to their class, which constituted a balanced dataset of 4394 examples.

The Dataset was then processed following the same methods as before. We first Tokenize the words and convert them to indexes and then were put into an embedding layer which used the GLoVe embedding weights with a 300 dimension.

Then, for the CNN and LSTM models, we padded the sentences to have the same length.

As for the SVM model, we used an GlobalMaxPooling function to have a matrix that could be given to the model. We tried using GlobalAveragePooling function but the other function gave better results.

7 Discussion and Results

We will evaluate the classifications on the following metric: F1-Score This scores measures accuracy using precision and recall. Precision is the ratio of true positives to all predicted positives. Recall is the ratio of true positives to all actual positives. Lets say the true positive is denoted by 'tp', false positive as 'fp', false negative as 'fn', precision as 'p' and recall as 'r'.

This metric gives equal weightage to both precision and recall and will try to maximize both precision and recall simultaneously. This would favor a moderately good performance on both over extremely good performance on one and poor performance on the other.

$$F1 = \frac{2pr}{p+r} \text{ where } p = \frac{tp}{tp+fp}, r = \frac{tp}{tp+fn}$$

Figure 6: F1 score

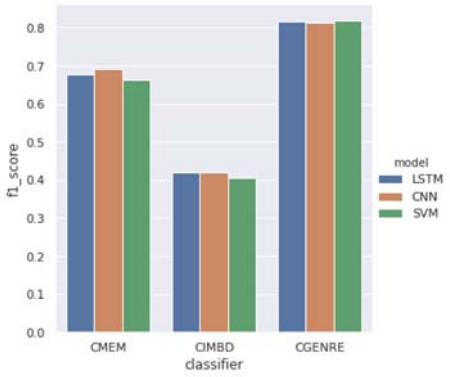

Figure 7: F1 scores on various parameters for all models

Taking a look at the F1 scores for each of the models on each classification, the Convolutional Neural Network performs better than the other two for CMEM classifier.

SVM however does surprisingly better than LSTM and CNN for CGENRE classifier .

This indicates that the Convolutional Neural Network seems to have learned the best representation of the dialogues data to be able to label it with its respective characteristics. Taking a look at the CNN loss function, we can see that even though the training loss decreases steadily, the loss for the validation stays the same and goes up near the end. This shows that the network is overfitting, and overall the labels do not seem to be classifiable by training on this data.

Taking a higher look at the task, it makes sense that trying to classify the speaker's gender based on a single utterance is a really hard task especially with no longer sentences or contextual data.

8 Memorability of dialogues based on gender:

As we mentioned in the introduction, we wanted to use deep learning and natural language processing for exploring the gender bias in Hollywood.

There have been studies and statistics that show that women are less represented, are paid less, and given less dialogues.

The ideal way to understand if the dialogues given to men and women differ in terms of memorability is if we could analyze a dataset that has the gender and

the memorability of every dialogue. However, we don't have such a dataset , but we do have a dataset of dialogues for which we know the gender and a different dataset of dialogues for which we know memorability. We propose to use the classification models previously built CGENRE to classify these dialogues to get the label (gender) that we don't have. We then used the CGENRE classifier on the data that is labeled with memorability. We noticed that we have 0.99 of memorable quotes that come from men and only 0.1 of memorable quotes that come from women.

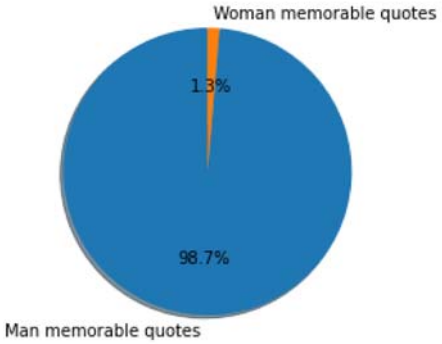

Figure 8: Memorable quotes

We also used the data with labeled gender to use the CMEM classifiers. We saw that most of the classifiers didn't show a big difference between genders and produced balanced results.

9 Conclusion

We used a variety of classification models to model the dialogue data to be able to classify various attributes of it. Each of the classifiers had varying degrees of success with the CNN outperforming the rest followed by the LSTM. Some labels were harder to predict than the others based on their nature. For example, predicting the IMDB rating of a movie based on a single dialogue is arguably a pretty impossible task even for humans as there are so many other factors that influence this. While gender in many other cases seems to be reasonably predictable, in this case, where we are analyzing individual utterances and long speeches or conversations and without any contextual information, we haven't been able to achieve high accuracy. We also ran multiple classifiers on the memorability dataset and while this was also a difficult characteristic to model without additional information, we achieved a reasonable F1 score with the CNN classifier.

We used these classifiers to then analyze the differences in the memorability of female and male quotes as a way to understand the qualitative gender

differences in Hollywood.

We saw that when using memorability classifiers on the gender known quotes, the differences in gender became less. This seems to indicate that at least according to the memorability dataset we trained on, there do not seem to major differences in the memorability of female and male quotes. However, the reverse analysis using gender classifiers on memorability known quotes did not yield any results most likely to the overall low accuracies of the gender classifiers.

10 Bibliography

1) https://towardsdatascience.com/classify-toxic-online-comments-with-lstm-and-g love-e455a58da9c7

2) https://towardsdatascience.com/what-the-heck-is-word-embedding-b30f67f01c81

3) https://machinelearningmastery.com/clean-text-machine-learning-python

4)https://towardsdatascience.com/introduction-to-natural-language-processing-for-text-df845750fb63

5)https://www.analyticsvidhya.com/blog/2019/04/predicting-movie-genres-nlp-multi-label-classification/