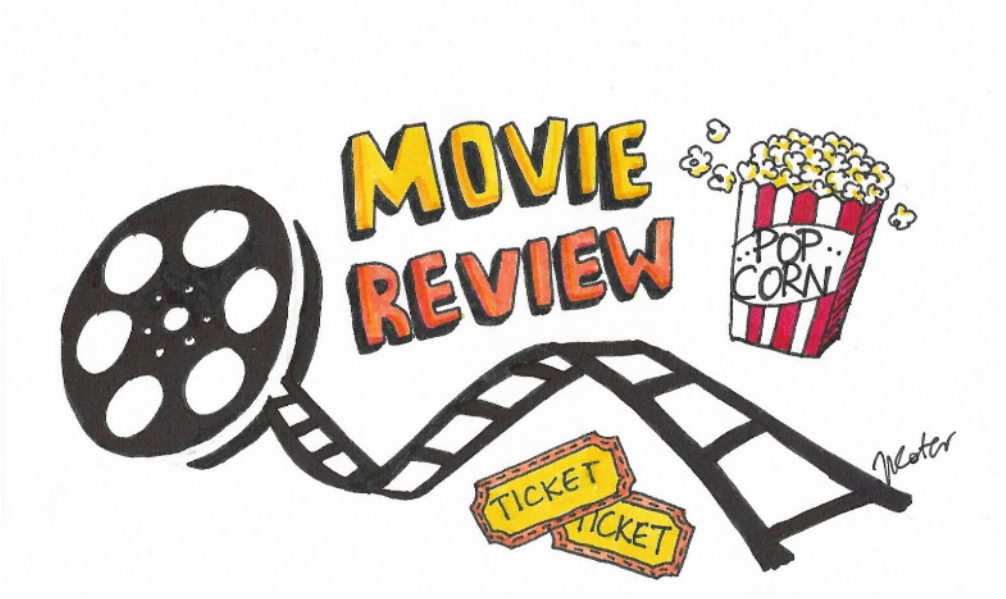
2022 Report for Homework 2 Knowledge Discovery Course - UNICT



Movies reviews sentiment analysis

Berlotti Mariaelena *mariaelenaberlotti.mb@gmail.com*
Di Grande Sarah *digrandesarah@gmail.com*
Licciardello Cristina *cris.licciardello@gmail.com*

1 Dataset Description

The object of our report is the analysis of the Movies reviews dataset available at the following link: https://www.kaggle.com/competitions/sentiment-analysis-on-movie-reviews/data The dataset contains 156060 movie reviews; in the following image it's possible to observe the data table structure:

	Phrase	Sentiment
0	A series of escapades demonstrating the adage ...	1
1	A series of escapades demonstrating the adage ...	2
2	A series	2
3	A	2
4	series	2
...	...	...
156055	Hearst 's	2
156056	forced avuncular chortles	1
156057	avuncular chortles	3
156058	avuncular	2
156059	chortles	2

156060 rows × 2 columns

Figure 1: Datatset structure

As we can observe, the dataset is made by two columns: 'Phrase', and 'Sentiment'. In particular, the 'Sentiment' is a categorical variable with five levels:

- **0** = 'negative',
- **1** = 'somewhat negative',
- **2** = 'neutral',

- **3** = 'somewhat positive',
- **4** = 'positive'.

In the following image, there is the representation of the levels distribution:

```
2    79582
3    32927
1    27273
4     9206
0     7072
Name: Sentiment, dtype: int64
```

Figure 2: Levels distribution

The problem we are dealing with is a sentiment analysis problem: starting from a movie review our RNN should be able to predict the sentiment associated.

2 Data Pre-processing

Before giving to the model our data, they need to be pre-processed. The first step of this part is the division of the data into inputs and labels; both the two have been changed into a text format. Next, we changed uppercase reviews into lowercase and we removed the punctuation since we want that the model focuses on the words, not on symbols. Next, we removed the last element for both input and labels since it is an empty string. Then, from the text format we extracted the single sentences. Each sentence can be seen as a list of characters with an average length of about 7 words. As a final outcome we have two lists: one containing all the sentences, and for each sentence we have a list containing its characters. The following step of our analysis was the creation of the vocabulary; indeed, we associated for each word a unique number, for a total of 16403 tokenized characters.

In order to have the same sentences length, we applied the zero padding; we decided to take as a reference sentences length the maximum one, that is 48 words. Before giving data to the model, we convert them into a tensor.

In the link provided above, two datasets were given: a labeled train dataset and an unlabeled test dataset. We decided to use the labeled one and to divide it into three parts.

- **training:** 124848 items
- **validation:** 15606 items
- **test:** 15606 items

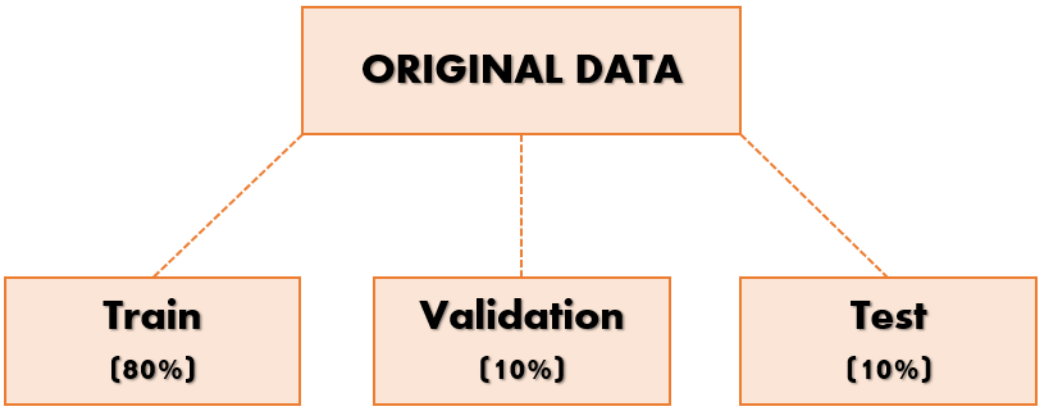


Figure 3: Dataset division

3 Model Description

We created an RNN network and we decided to evaluate its performances adding/subtracting layer; in particular, we started from a simpler network till we arrived to a deeper one.

3.1 Network structure

Our network is composed by 1 layer and a classifier. The model takes as input an embedding vector with a size of 400. Then, this vector is passed

to a Long Short Term Memory (LSTM) cell; principally LSTM contains a memory state to hold information and mechanisms for adding and discarding information during the learning process. This memory is shown as a cell state running through the LSTMcell and is altered through element-wise operations, as multiplication and sum. The amount and which data to let through is selected by various gates in the cell. Fundamental components of an LSTM cell are a forget gate, input gate, output gate and a cell state. The forget part is computed using a multiplication. The cell state represents the store part where information are maintained. For each time step we update the state, summing all the information to keep. Finally, the output gate computes the final output using the updated cell state. The final output of the LSTM cell is passed to the linear classifier that returns back the classification of the reviews according to the sentiment.

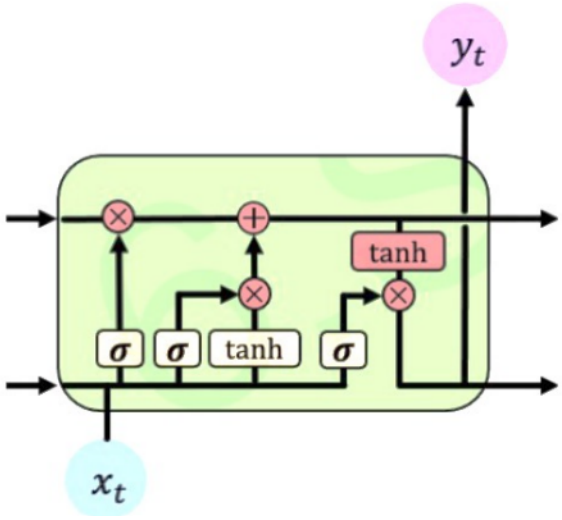


Figure 4: Network structure

4 Training procedure

With a batch size of 250 and the Cross Entropy loss, we decided to apply the Adam optimizer with a learning rate of 0.001 and a weight decay of 0.0005.

5 Performance analysis

Running the code for 20 epochs, we got a train accuracy of 0.71, a validation accuracy of 0.66 and a test accuracy of 0.65, while the loss functions reach the following values: TrL= 0.70, VL=0.82, TeL=0.82.

6 Confusion matrix

Figure 5: Confusion matrix

According to what is shown by the confusion matrix we can see that:

- For the category **0** the model predicts in the correct way 0.44 of the true labels.
- For the category **1** the model predicts in the correct way 0.59 of the true labels.

- For the category **2** the model predicts in the correct way 0.76 of the true labels.
- For the category **3** the model predicts in the correct way 0.52 of the true labels.
- For the category **4** the model predicts in the correct way 0.54 of the true labels.

7 Ablation studies

We tested the performances of this first network also in the following cases:

- **1st + 2nd + 3rd + 4th + 5th layers + Classifier:** in this case we obtained the following accuracy: TrA=0.70, VA=0.66, TeA=0.66.
- **1st + 2nd + 3rd + 4th layers + Classifier:** in this case we obtained the following accuracy: TrA=0.71, VA=0.65, TeA=0.66.
- **1st + 2nd + 3rd layers + Classifier:** in this case we obtained the following accuracy: TrA=0.70, VA=0.67, TeA=0.67.
- **1st + 2nd layers + Classifier:** in this case we obtained the following accuracy: TrA=0.71, VA=0.67, TeA=0.67.

8 Conclusions

Once trained the different networks, we decided to choose the one with a single layer because that returned 0.65 of accuracy in 20 epochs. As we have seen, this model is not the best one in terms of performances; indeed, the network with two layers gave us an higher test accuracy. However, since the difference between the two accuracy is low, we have decided to choose the simplest model.