Emotion Detection

Barbu Iulia-Andreea, Pavalasc Irina

Faculty of Computer Science, University of Bucharest

Introduction

Deep Learning has proved itself indispensable in a variety of domains, including natural language processing. With the rise of social media and text-based communication, the demand for sentiment analysis is greater than ever. The purpose of this paper is to analyse a series of dialogues with the intent of predicted the conveyed emotion. This is a rather difficult task, given the complexity of human emotion, which makes analysing the context of an utterance extremely important.

Dataset

The data set used is EmoContext, which contains a series of textual dialogues in English. As the context can be crucial when analysing the emotions conveyed in text, each dialog contains three utterances. The data has been gathered from user interaction with a conversational agent and were annotated for four different emotion classes - Happy, Sad, Angry and Others.

- The training data set contains 30160 dialogues.
- The test data sets contains 5509 (Test2) dialogues.

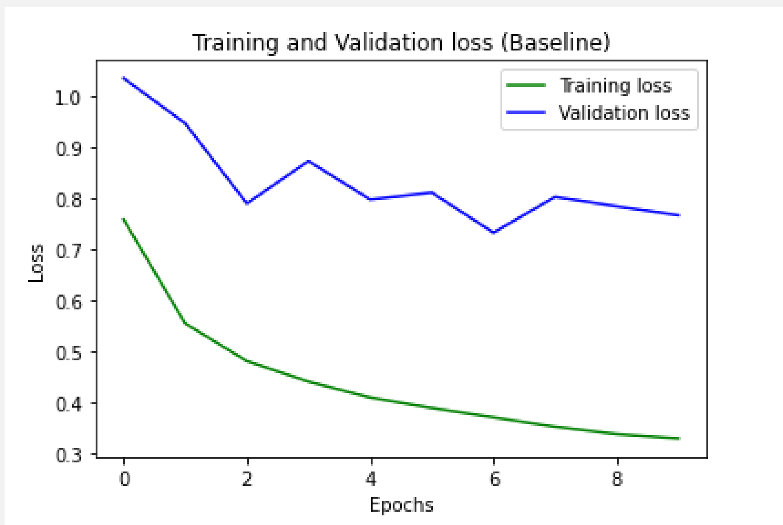
40	I don't no	you just said you did!	What	others
41	Send me your nekde pic	after this battle, sure :D	Let's start	others
42	OK friend's	YA YA VERY GOOD FRIEND	Very nice friends	happy
43	She is ignoring me	no im not ignore you	😱😱😱	sad
44	Happy	I'm never happy. But I'm almost happy.	Hij to the top	happy
45	Nothing	I'm just being nice to you.	In ur school?	others
46	Not sure	let me know!	Pls you tell	others
47	I like your positive approach	No problem, glad to help	😀😀	others
48	You tell me	Tell you what?	My name	others
49	Hmmm	I can talk now! :D	U don't know what hppnd	others
50	What I did?	You didn't Asmita, see my answer	😂😂	happy
51	About?	I never said that it's freezing.	Ya you didn't	others

Data Processing

- Emoji can be considered a big part of text-based communication so we took the approach of replacing them with their corresponding descriptions using the demoji package.
- We used the spacy library to 'tokenize' our data.
- For the word embeddings we used GloVe, an unsupervised learning algorithm for obtaining vector representations for words.

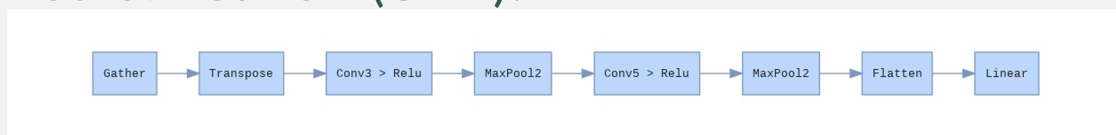
Loss Function

For the loss function we used Cross Entropy Loss. Because the data is unbalanced(most of the dialogues are annotated as 'others'), we added the weights [0.33, 1. , 1. , 1.].

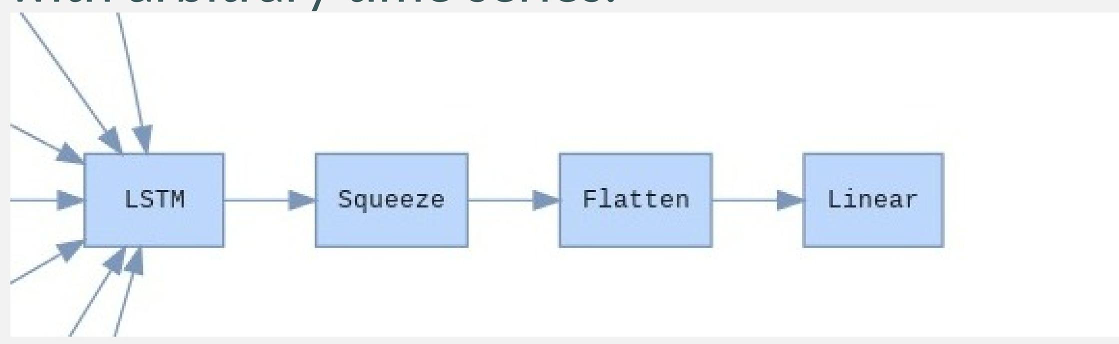


Neural Network Arhitecture

Baseline Model - we used a convolutional neural network(CNN).

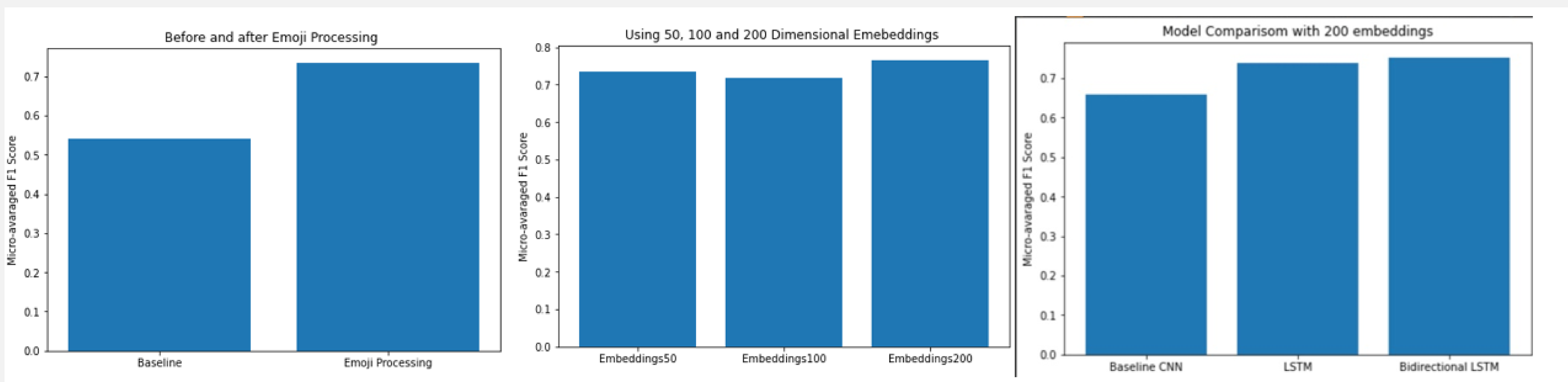


Long short-term memory (LSTM) is a Recurrent Neural Network (RNN) capable of processing sequential information.LSTM networks are suitable for classifying, processing, and making predictions based on time series data, as there may be gaps of unknown duration between important events in a time series, and they work with arbitrary time series.



Bidirectional LSTM - instead of training a single model, we introduce two. The first model learns the sequence of the input provided, and the second model learns the reverse of that sequence. Since we do have two models trained, we need to build a mechanism to combine both. It is usually referred to as the Merge step.

Results



Experiments

We used a micro-averaged F1 score as a metric for our models' performance.

Dimensional Embeddings - we used our baseline model to plot the difference in performance between using 50, 100 and 200 dimensional embeddings. The results show a slight improvement when using 200 dimensional embeddings.

Emoji Processing - we used our baseline model to plot the difference in performance when processing the emoji found in our dialogues. We noticed a significant improvement.

Baseline vs RNN vs Bidirectional RNN - we compared the performance between our three models.

- In the unidirectional RNN model we also ran with 2 LSTM instances, but the differences in results were not noticeable.

Conclusion

Even though the LSTM is very efficient in NLP, for sequential data processing, and the CNN is designed rather for images, in this matter of sentiment analysis it turns out that the CNN manages to perform almost as well, and the intuitive reason would be that the results are successful when the model focuses on groups of suggestive keywords and not necessarily on many bindings between sequences.

References

[1] Joshi Agrawal Chatterjee, Narahari.
Semeval-2019 task 3: Emocontext contextual emotion detection in text.
Microsoft.