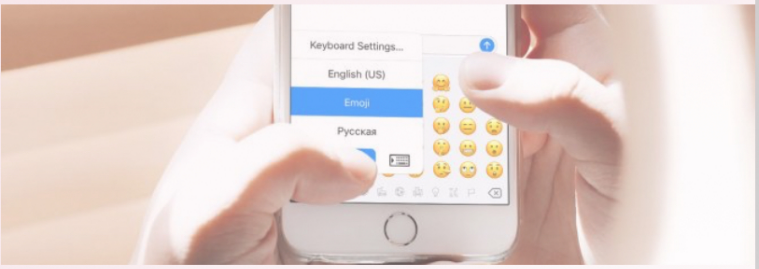
Emoji Predictor: Introduce emoji embedding to Emoji Semantics Modelling

Qiqi Zeng Su Shen Lulu Wan
COMP 0087 Natural Language Processing Project



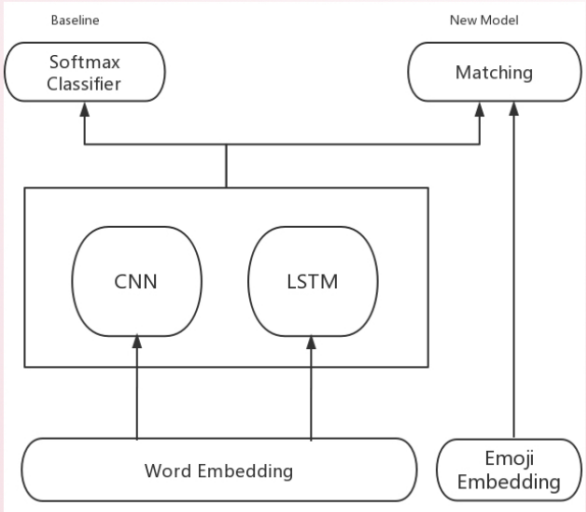
Demo

I got an offer from University College London. 😊
A Boeing 737 Max8 airplane crashed in Ethiopian. 😱
I heard someone expressed discriminating opinions. 😠
I struggle with countless deadlines these days. 😢
A couple abused animal after drunk. 😒
I said something bad behind my classmate's back. 😳
He has wasted too much time on watching soap opera. 😒

Introduction

The boom of emoji usage stimulates the study of semantics emojis from an NLP standpoint. In this project, we first applied CNN and LSTM to generate the embedding of the sentence, and then matched it with the emoji embedding (emoji2vec) through cosine similarity to find the most corresponding emoji.

Model Structure Overview



Build Emoji Prediciton model by introducing emoji embedding (emoji2vec) to Neural Network models (LSTM and CNN)

Methods

Baselines

Long Short Term Memory (LSTM)
Convolutional Neural Network (CNN)

Layers

Word Embedding Layer	Get the distributed representation of each word.
Convolutional Layer	Extract n-gram feature of a sequence.
Pooling Layer	Synthesize local embeddings into one vector.
Hidden Layer	Apply a linear transformation.
Emoji Embedding Layer	Find emoji corresponding vectors with emoji2vec
Matching Layer	Match emoji embedding with sentence embedding.

Training Algorithm

Input	One sentence, its correct emoji(label)
Output	Updating model parameters
Implementation	

For each emoji x_i do:
1. Forward propagation to calculate the matching scores between the vector of the given sentence and each emoji vector
2. Calculate the loss using the CrossEntropyLoss function
3. Backward propagation to update model parameters using SGD method.

Evaluation Metrics

on test dataset (1486)

Metrics	Accuracy	MacroF1	Recall	Precision
LSTM Baseline	0.35	0.33	0.33	0.33
CNN Baseline	0.43	0.40	0.41	0.42
LSTM with emoji2vec	0.40	0.32	0.34	0.40
CNN with emoji2vec	0.60	0.58	0.61	0.63

Conclusion

Empirical results on testing dataset demonstrate that our approach significantly improves both the accuracy and efficiency compared to traditional classifiers. .