EmojiPred – Emoji Prediction on the Fly

Varun Khurana
varun19124@iiitd.ac.in

Harsh Kumar
harsh19043@iiitd.ac.in

Jahnvi Kumari
jahnvi19469@iiitd.ac.in

1 Introduction

With the growth of social media and text messaging platforms, emojis are becoming increasingly popular to express emotions. But, very few keyboards have been developed and deployed which help users choose emojis while typing. Moreover, to the best of our knowledge, no such recommendation system has been developed for Indian languages. In this paper, we propose a deep learning based emoji recommendation system – **EmojiPred**, which can be used for on-the-fly prediction of emojis while typing words from English and a few Indian languages including Hindi, Bengali and Telugu. EmojiPred leverages a combination of both token embeddings and contextual sentence embeddings to appropriately capture semantics and context, for achieving best results. Extensive experiments demonstrate the performance of our model. We have deployed the keyboard on https://emoji-pred.vercel.app/.

2 Related Work

There are numerous ways in which the problem of Emoji Prediction can be modelled.

Guibon et al. (2018) used a private text-message corpus to form features using bag of words (BoW) and n-grams. They used MultiRandom forest for predicting. Urabe et al. (2013) developed an Eastern emoticon prediction system.

Illendula and Yedulla (2018) explored the use of CNN classifier and Knowledge concepts to predict emojis, after generating text embedding of images.

Felbo et al. (2017) trained a deep learning model based on LSTMs and weighted attention to recommend emojis. This work also proposed transfer learning using 'chain-thaw' mechanism.

Mathew (2020) used DeepMoji for transfer learning and neural networks to build a smart reply and emoji prediction system.

3 Pre-processing

3.1 Datasets

For emoji prediction for English, we used the Twitter Emoji Prediction [1] dataset. It comprises of 70,000 anonymised tweets containing to 20 unique emojis. For Indian languages, we used the Twitter Corpus for low resource languages for sentiment analysis and emoji prediction (Singh and Choudhary, 2018). It consists of cleaned tweets in Hindi, Bengali and Telugu languages.

3.2 Data Augmentation

Due to the limited availability of labelled tweets, we performed data augmentation using Backtranslation (Poncelas et al., 2018). It is a popular technique for generating paraphrases where in the source language text is translated to an intermediate language, which is further translated back to the source language. In our case, we performed Backtranslation on the English tweets corpus, using French as the intermediate language. For this purpose we used pre-trained transformer based models using Hugging Face [2] library.

3.3 Cleaning

Data was cleaned using regular expressions. All punctuation marks were removed. All numbers, special characters, user tags and hashtags were removed. We explored the possibility of keeping hashtags intact and proceeding further, but more often than not the sense of hashtags was difficult to judge. Any extra spaces were also removed. The text was converted to lower case, and then tokenized. All stopwords were removed from the list of tokens.

We fixed the size of input size to be 10. For tweets having more than 10 valid tokens, we con-

[1] https://www.kaggle.com/hariharasudhanas/twitter-emoji-prediction

[2] https://huggingface.co/

❤️	😊	😍	😌	😐	😠	😢	🤔	😡
9.54	9.51	9.42	8.58	7.72	7.50	7.22	6.78	6.73
😭	💕	😉	😁	💙	😘	😶	😜	💜
4.92	4.32	3.82	2.90	2.81	2.56	2.33	1.87	1.46

Table 2: Percentage of emojis in Hindi tweets

😊	❤️	😍	😐	😠	😌	😘	😢	🤔
15.65	13.99	10.46	10.22	8.66	8.59	7.44	5.80	5.70
😶	😭	😡	😁	😜	😉	💕	💙	💜
4.64	3.66	1.32	1.08	1.04	0.67	0.52	0.49	0.07

Table 3: Percentage of emojis in Bengali tweets

😊	😠	😍	❤️	😐	🤔	😏	😢	😭
22.67	16.55	13.38	10.77	6.68	5.67	5.05	4.19	3.91
😡	😜	😁	😉	💕	😶	😘	💜	💙
3.01	2.29	1.57	1.36	0.94	0.84	0.54	0.31	0.25

Table 4: Percentage of emojis in Telugu tweets

Figure 1: Caption

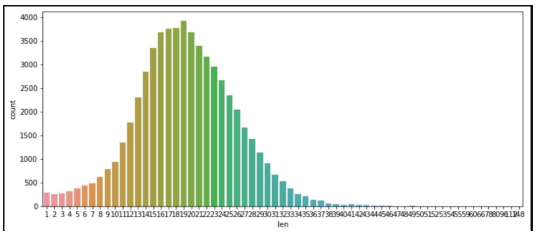

Figure 2: Length of Hindi tweets before cleaning, mean length = 17 tokens

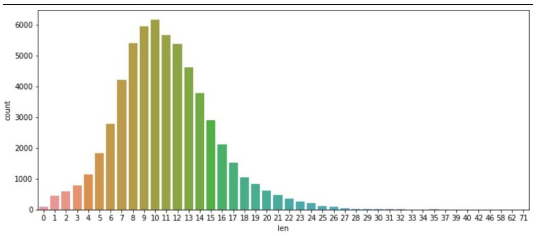

Figure 3: Length of Hindi tweets after cleaning, mean length = 10 tokens

sidered the last 10 tokens for further processing. For tweets having less than 10 tokens, we padded them with zero valued vectors. This ensured that it does not affect the overall sentiment of the tweet.

Tweets having less than 2 valid tokens, or more than 20 valid token were ignored. It was noticed that for smaller tweets, the text was majorly noise, and was not contributing much significant idea about the emoji. Similarly, for a very long text, the emoji was harder to predict, as it was very random in nature. This technique was specially helpful in reducing the size of translated Hindi tweets. We observed that translated tweets were larger in size. After removing of irrelevant tokens and removing stopwords, the size was ideal for further processing.

4 Methodology

4.1 Embeddings

EmojiPred utilises both token embeddings and sentence embeddings. The Indian languages Hindi, Bengali and Telugu were also translated to English using Google Translate API [3] for further processing. This is because of lack of availability strong word embedding models for low-resource languages.

For preparing token representations, 200-dimensional GloVe embeddings (Pennington et al., 2014), pretrained on a vocabulary size of 1.2 million words on twitter corporus were used, which is ideal for our task. Since we are limiting to only 10 tokens per tweet, so each tweet is encoded as a 10×200 matrix. On the other hand, Universal Sentence Encoder (Cer et al., 2018) was used to encode sentences into 512-dimensional contextual embedding vectors.

4.2 Model Architecture

Each language has a separate independent model for it's corresponding set of tweets. This is necessary the nature of these datasets is completely different. Each of them have different set of emojis labels. Also, emotion is very personal. It is very possible to have the same emoji convey different emotion in context of different languages. To overcome this, separation of models is important.

EmojiPred implements two parallel pipelines which are integrated together later stage to yield the final predictions. The first one runs at token level, while the latter works at sentence level as illustrated by Figure 4.

4.2.1 Token-level pipeline

In the first pipeline, word level embedding is used. Here, it's important to note that the embedding used are the same for all the language models. For language other than English, the text is first cleaned and converted to English before finding the word embedding. The embedding are passed on to spatial dropout layer. This helps in regularisation and this prevents overfitting of the model.

After this, the output is fed to two Bi-directional LSTM layers. The rationale is that BiLSTMs are

[3] https://cloud.google.com/translate

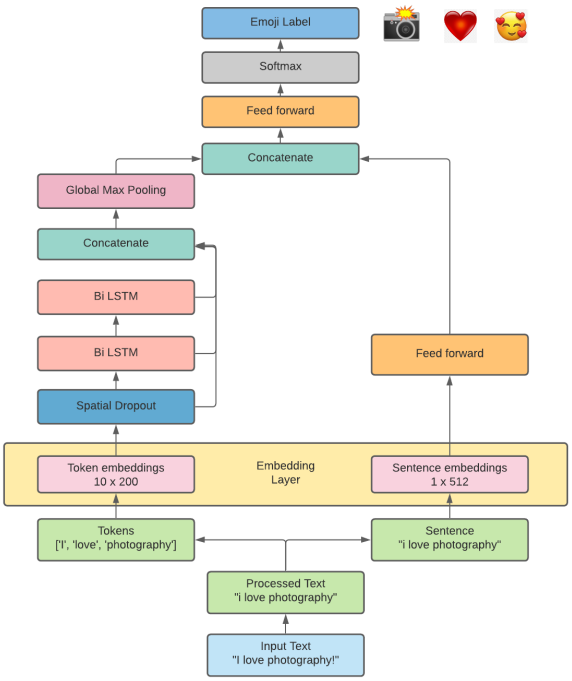

Figure 4: Model architecture

able to capture the context very well in both directions compared to other sequential models. We implement skip connections by concatenating output of the previous three consecutive layers as it helps in avoiding loss of information from previous stages and also tides over the problem of vanishing gradients. The result is then passed to Global Max Pooling stage to reduce the output from 3 dimensions to 2 dimensions.

4.2.2 Sentence-level pipeline

The 512-dimensional Universal Sentence Encoder based sentence embeddings are fed to a feed-forward network for dimensionality reduction of the vectors. The 1×512 sentence encodings are reduced to the size of 1×200.

4.2.3 Prediction Layer

The results from these two parallel pipelines is concatenated and is fed to a fully connected layer where softmax actiavtion function is applied to yield the probability of each emoji label. The label corresponding to maximum probability is served as the final prediction of the model.

During the training phase, we have used Categorical Cross Entropy as the loss function.

Model	Accuracy
Multinomial Naive Bayes	0.24
SVM	0.40
Multilayer Perceptron	0.57
Causal Convolutions	0.66
EmojiPred	**0.72**

Table 1: Comparison of EmojiPred results against other models on the English dataset.

5 Experiments

5.1 Baselines

We have demonstrated the performance of our model by comparing against a number of baseline models.

- **Machine Learning based models:** such as Multinomial Naive Bayes and Decision Tree based on term frequency–inverse document frequency vectors. These are simple and intuitive models, however perform poorly in the given task as they fail to capture the context of the sentences.

- **Deep Learning based models:** such as Multilayer Perceptron and Causal Convolutions. MLPs are good function approximators but are prone to overfitting. They do not perform well for sequential data. In temporal or causal convolutions, the output at instant t only depends on inputs at instants $< t$. They are highly suitable in applications using temporal data. However, they are only able to accumulate information in one direction, unlike BiLSTMs which limits their use in other applications. Though better than Machine Learning models, the results of these deep learning models lag behind or are comparable to those of EmojiPred.

5.2 Metrics

$$Accuracy = \frac{TP + FP}{TP + FP + TN + FN} \quad (1)$$

Top-5 Accuracy: If any of the model's top 5 predictions match the ground truth label, it is considered an acceptable prediction. Precision

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

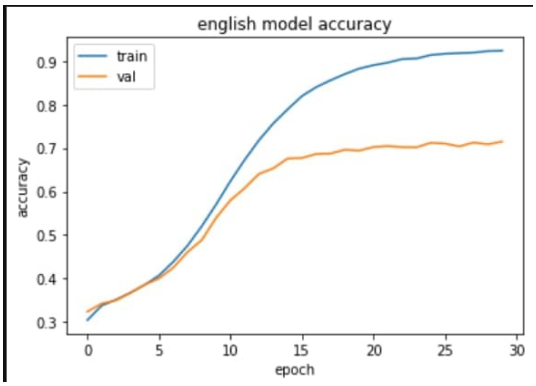

Figure 5: Accuracy curves for EmojiPred English model.

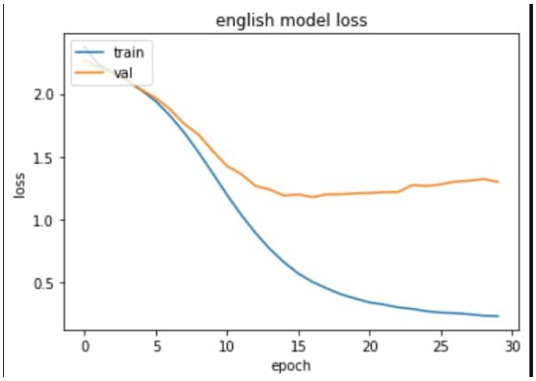

Figure 6: Loss curves for EmojiPred English model.

F1-score

$$F1 - score = \frac{2PR}{P+R} \quad (4)$$

6 Results and Analysis

6.1 Comparison against other models for English dataset

Table 1 shows the performance of EmojiPred for prediction of emojis for English language, compared against a number of baseline models. It can be observed that EmojiPred performs better than all baseline models.

6.2 Performance on English, Hindi, Bengali and Telugu datasets

Table 2 exhibits the results of EmojiPred model tested on English, Hindi, Bengali and Telugu.

It can be observed that the performance on Indian languages is not up to the mark. They are low-resource languages. We translated them to English to obtain their token-level and sentence-level embeddings. Since, translation cannot be perfectly accurate, so some information might have been lost in the translation phase.

6.3 Ablation Study

Table 3 highlights the contribution of some components in the overall performance of EmojiPred, when evaluated on English dataset.

- **Universal Sentence Encoder (USE):** The parallel sentence level embeddings pipeline boosts accuracy as it captures the overall semantic meaning and context of a sentence.
- **BiLSTM layers:** BiLSTM layers help to capture the semantics and context in both forward and backward directions.
- **Spatial Dropout:** They are helpful in regularisation, thereby preventing model overfitting and increasing model generalisability.

7 Conclusion

EmojiPred is a fairly well-performing multi-lingual emoji recommendation model. However, it has much scope of further improvement. In future, use of attention to compute the relative importance of words vis-a-vis each other can be considered, since some words contribute more significantly to the emotion represented by a sentence than others.

Due to limited computational resources, we could not train our models on larger datasets, which is imperative to achieve higher generalisability. Effieicent and accurate Machine Translation models can be developed to minimise loss of information incurred during the translation phase for Indian languages.

References

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. pages 1615–1625.

Gaël Guibon, Magalie Ochs, and Patrice Bellot. 2018. Emoji recommendation in private instant messages. SAC '18, page 1821–1823, New York, NY, USA. Association for Computing Machinery.

Anurag Illendula and Manish Reddy Yedulla. 2018. Learning emoji embeddings using emoji co-occurrence network graph. *ArXiv*, abs/1806.07785.

Language	Accuracy	Top-5 Accuracy	Precision	Recall	F1-score
English	0.72	0.77	0.69	0.66	0.67
Hindi	0.17	0.59	0.17	0.17	0.14
Bengali	0.23	0.68	0.13	0.23	0.14
Telugu	0.42	0.76	0.43	0.42	0.41

Table 2: Results on the various languages.

Model	Accuracy
EmojiPred	**0.72**
EmojiPred – USE	0.69
EmojiPred – Spatial Dropouts	0.64
EmojiPred – BiLSTM	0.44

Table 3: Importance of components of EmojiPred towards overall performance.

Steffan Mathew. 2020. On device deep neural networks for emoji and reply prediction.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger, and Peyman Passban. 2018. Investigating backtranslation in neural machine translation.

Rajat Singh and Nurendra Choudhary. 2018. Twitter corpus of Resource-Scarce Languages for Sentiment Analysis and Multilingual Emoji Prediction.

Yuki Urabe, Rafal Rzepka, and Kenji Araki. 2013. Emoticon recommendation system for effective communication. In *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)*, pages 1460–1461.