# SeerNet at EmoInt-2017: Tweet Emotion Intensity Estimator

Venkatesh Duppada, Sushant Hiray
SeerNet Technologies, LLC

DeepAffects

## Overview

**Problem Statement**
The Task 1 of WASSA-2017 poses the problem of finding emotion intensity of tweets given an emotion. This task focuses on finding emotion intensity (0.0 to 1.0) of four emotions namely anger, sad, joy, fear.

**Approach**
We pre-process the tweets and create sentence level embeddings using lexicons and word vectors. After performing feature extraction, we applied various regressors like AdaBoost, GradientBoost to maximise Pearson's correlation coefficient. Finally an ensemble is created by choosing best performing models.

**Results**

- Third in best Pearson correlation coefficient (official)
- Second in best Pearson correlation coefficient for emotion intensity greater than 0.5

## Features

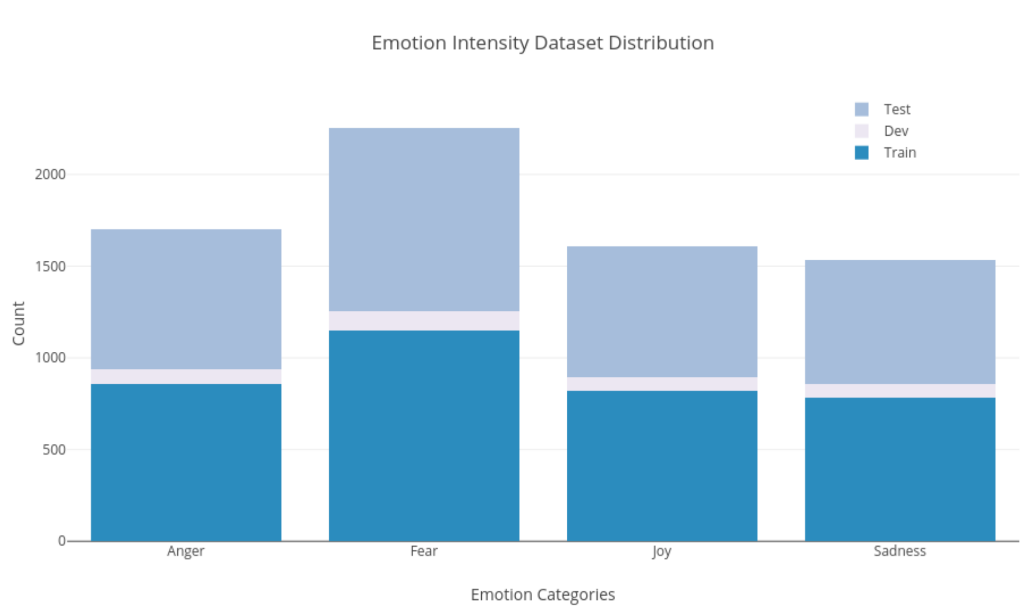
In our approach we converted a tweet to a sentence embedding using three approaches:

Lexicon based
- Lexicons associate words to corresponding sentiment or emotion metrics.

Word Vector based
- Semantic relationship between words are represented using low dimensional feature vectors.

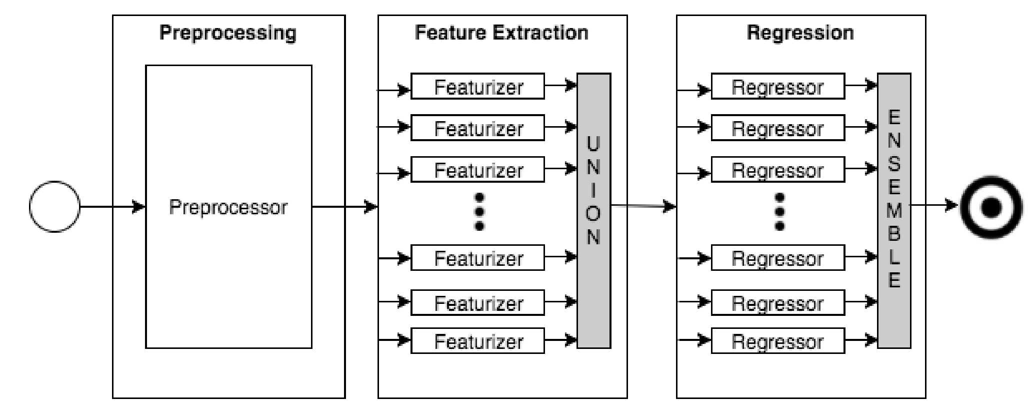
Emoji Vector based
- Semantic relationship between emojis are represented using low dimensional feature vector.

## EmoInt Dataset




## System Description

**Preprocessing**
- Tweet aware tokenizer to extract meaning fun tokens like emoticons, emojis, punctuations etc.
- Replace unnecessary tokens with standard notations.
  - URLs to URL
  - Numbers to NUMBER
  - Times to TIME
  - Usernames to USERNAME etc.




SeerNet EmoInt System Architecture

**Feature Extraction**

We have used all well known lexicons and collected the metrics on sentence level.
For example
- Bing Liu Opinion Lexicon [1]
  - Average positive and negative sentiment of words in a tweet
- NRC Affect Intensity [2]
  - Average emotion intensity of words for emotion categories in a tweet
- Average number of negation words etc.

Similarly on word/emoticon vector side we used the following.
- GloVe Embeddings [3]
- Edinburgh Embeddings [4]
- Emoji Embeddings [5]

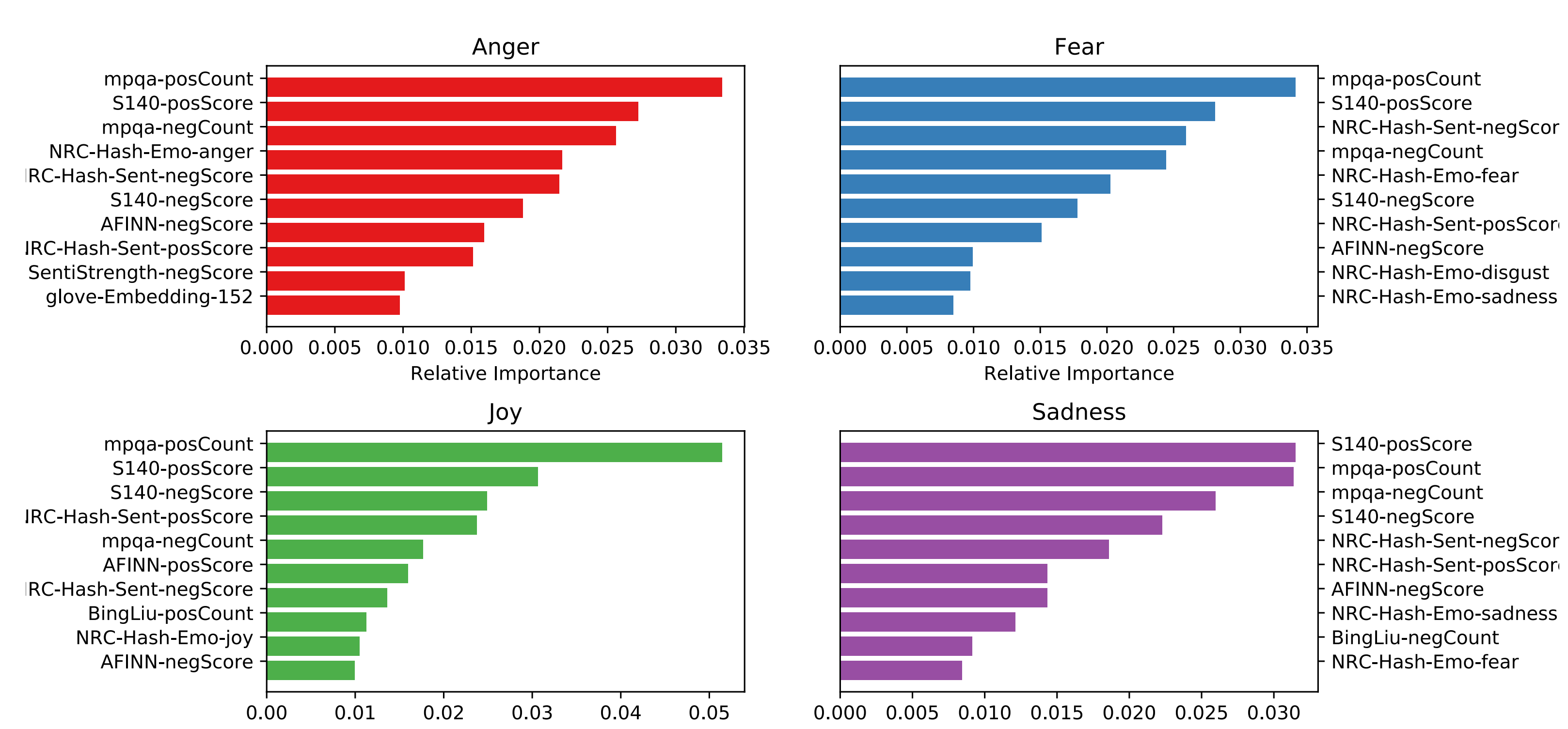
The final feature vector is the concatenation of all the individual features.

**Training**

- Perform 10 fold cross validation on dev + train data
- Trained regressors like AdaBoost, GradientBoost and RandomForests etc.
- Select best models on cross validation minimising Pearson Correlation Coefficient
- Create an ensemble of best performing methods

## Results

- Best results are obtained on ensemble created using best performing models across all emotion categories.
- Best Pearson correlation coefficients across all the emotion categories on test data.
  - Anger - 0.715183
  - Fear - 0.702265
  - Joy - 0.55209
  - Sadness - 0.530501
- Below is the top 10 feature importances of the features used in finding emotional intensity.




## Limitations

The following are some of the major limitations of our system.

- The system sometimes has difficulties in capturing the overall sentiment due to presence of words misleading intensity emotion and this leads to amplifying or vanishing intensity signals.
  - @MannersAboveAll *laughs louder this time, shaking my head*That was really cheesy, wasn't it?
    - Gold Intensity - 0.083
    - Predicted Intensity - 0.4936
- The system also fails in predicting sentences having deeper emotion and sentiment which humans can understand with a little context.
  - Ibiza blues hitting me hard already wow
    - Gold Intensity - 0.833
    - Predicted Intensity - 0.4247
  - Here tweet refers to post travel blues which humans can understand but with little context, it is difficult for the system to accurately estimate the intensity.

## Conclusions & Future Work

**Conclusions**
- The paper studies the effectiveness of various affect lexicons word embeddings to estimate emotional intensity in tweets.
- A light-weight easy to use affect computing framework to facilitate ease of experimenting with various lexicon features for text tasks is open-sourced.
- Generic features which will be useful in other affective computing tasks on social media text not just tweet data.
- Good run-time performance during prediction, future work to benchmark the performance of the system can prove vital for deploying in a real-world setting.

**Future Work**
- Few problems explained in the analysis section can be resolved with the help of sentence embeddings which take the context information into consideration.

## References


- [1] Minqing Hu and Bing Liu. 2004. Mining and summarising customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pages 168-177
- [2] Saif M Mohammad. 2017. Word affect intensities. arXiv preprint arXiv: 1704.08798
- [3] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In EMNLP. volume 14, pages 1532-1543
- [4] Felipe Bravo-Marquez, Eibe Frank, and Bernhard Pfahringer. 2015. From unlabelled tweets to twitter specific opinion words. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM pages 743-746
- [5] Ben Eisner, Tim Rocktaschel, Isabelle Augenstein, Matko Bosnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. arXiv preprint arXiv:1609.08359