Seernet at EmoInt-2017: Tweet Emotion Intensity Estimator

Venkatesh Duppada and **Sushant Hiray**
Seernet Technologies, LLC
{venkatesh.duppada, sushant.hiray}@seernet.io

Abstract

The paper describes experiments on estimating emotion intensity in tweets using a generalized regressor system. The system combines lexical, syntactic and pre-trained word embedding features, trains them on general regressors and finally combines the best performing models to create an ensemble. The proposed system stood 3rd out of 22 systems in the leaderboard of WASSA-2017 Shared Task on Emotion Intensity.

1 Introduction

Twitter, a micro-blogging and social networking site has emerged as a platform where people express themselves and react to events in real-time. It is estimated that nearly 500 million tweets are sent per day [1]. Twitter data is particularly interesting because of its peculiar nature where people convey messages in short sentences using hashtags, emoticons, emojis etc. In addition, each tweet has meta data like location and language used by the sender. It's challenging to analyze this data because the tweets might not be grammatically correct and the users tend to use informal and slang words all the time. Hence, this poses an interesting problem for NLP researchers. Any advances in using this abundant and diverse data can help understand and analyze information about a person, an event, a product, an organization or a country as a whole. Many notable use cases of the twitter can be found here[2].

Along the similar lines, **The Task 1 of WASSA-2017** (Mohammad and Bravo-Marquez, 2017c) poses a problem of finding emotion intensity of four emotions namely anger, fear, joy, sadness from tweets. In this paper, we describe our approach and experiments to solve this problem. The rest of the paper is laid out as follows: Section 2 describes the system architecture, Section 3 reports results and inference from different experiments, while Section 4 points to ways that the problem can be further explored.

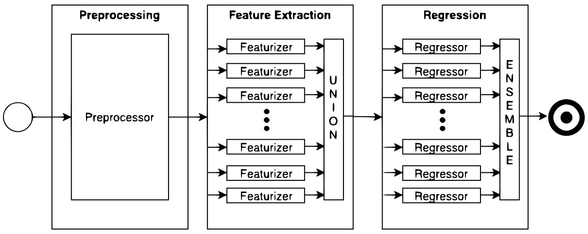

Figure 1: System Architecture

2 System Description

2.1 Preprocessing

The preprocessing step modifies the raw tweets before they are passed to feature extraction. Tweets are processed using **tweetokenize** tool[3]. Twitter specific features are replaced as follows: username handles to `USERNAME`, phone numbers to `PHONENUMBER`, numbers to `NUMBER`, URLs to `URL` and times to `TIME`. A continuous sequence of emojis is broken into individual tokens. Finally, all tokens are converted to lowercase.

2.2 Feature Extraction

Many tasks related to sentiment or emotion analysis depend upon affect, opinion, sentiment, sense and emotion lexicons. These lexicons associate words to corresponding sentiment or emotion metrics. On the other hand, the semantic meaning of words, sentences, and documents are preserved

[1] https://en.wikipedia.org/wiki/Twitter

[2] https://en.wikipedia.org/wiki/Twitter_usage

[3] https://www.github.com/jaredks/tweetokenize

and compactly represented using low dimensional vectors (Mikolov et al., 2013) instead of one hot encoding vectors which are sparse and high dimensional. Finally, there are traditional NLP features like word N-grams, character N-grams, Part-Of-Speech N-grams and word clusters which are known to perform well on various tasks.

Based on these observations, the feature extraction step is implemented as a union of different independent feature extractors (featurizers) in a light-weight and easy to use Python program EmoInt [4]. It comprises of all features available in the baseline model (Mohammad and Bravo-Marquez, 2017a) [5] along with additional feature extractors and bi-gram support. Fourteen such feature extractors have been implemented which can be clubbed into 3 major categories:

- Lexicon Features
- Word Vectors
- Syntax Features

Lexicon Features: AFINN (Nielsen, 2011) word list are manually rated for valence with an integer between -5 (Negative Sentiment) and +5 (Positive Sentiment). Bing Liu (Hu and Liu, 2004) opinion lexicon extract opinion on customer reviews. +/-EffectWordNet (Choi and Wiebe, 2014) by MPQA group are sense level lexicons. The NRC Affect Intensity (Mohammad, 2017) lexicons provide real valued affect intensity. NRC Word-Emotion Association Lexicon (Mohammad and Turney, 2010) contains 8 sense level associations (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and 2 sentiment level associations (negative and positive). Expanded NRC Word-Emotion Association Lexicon (Bravo-Marquez et al., 2016) expands the NRC word-emotion association lexicon for twitter specific language. NRC Hashtag Emotion Lexicon (Mohammad and Kiritchenko, 2015) contains emotion word associations computed on emotion labeled twitter corpus via Hashtags. NRC Hashtag Sentiment Lexicon and Sentiment140 Lexicon (Mohammad et al., 2013) contains sentiment word associations computed on twitter corpus via Hashtags and Emoticons. SentiWordNet (Baccianella et al., 2010) assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity. Negation lexicons collections are used to count the total occurrence of negative words. In addition to these, SentiStrength (Thelwall et al., 2010) application which estimates the strength of positive and negative sentiment from tweets is also added.

Word Vectors: We focus primarily on the word vector representations (word embeddings) created specifically using the twitter dataset. GloVe (Pennington et al., 2014) is an unsupervised learning algorithm for obtaining vector representations for words. 200-dimensional GloVe embeddings trained on 2 Billion tweets are integrated. Edinburgh embeddings (Bravo-Marquez et al., 2015) are obtained by training skip-gram model on Edinburgh corpus (Petrovic et al., 2010). Since tweets are abundant with emojis, Emoji embeddings (Eisner et al., 2016) which are learned from the emoji descriptions have been used. Embeddings for each tweet are obtained by summing up individual word vectors and then dividing by the number of tokens in the tweet.

Syntactic Features: Syntax specific features such as Word N-grams, Part-Of-Speech N-grams (Owoputi et al., 2013), Brown Cluster N-grams (Brown et al., 1992) obtained using TweetNLP [6] project have been integrated into the system.

The final feature vector is the concatenation of all the individual features. For example, we concatenate average word vectors, sum of NRC Affect Intensities, number of positive and negative Bing Liu lexicons, number of negation words and so on to get final feature vector. The scaling of final features is not required when used with gradient boosted trees. However, scaling steps like standard scaling (zero mean and unit normal) may be beneficial for neural networks as the optimizers work well when the data is centered around origin.

A total of fourteen different feature extractors have been implemented, all of which can be enabled or disabled individually to extract features from a given tweet.

2.3 Regression

The dev data set (Mohammad and Bravo-Marquez, 2017b) in the competition was small hence, the train and dev sets were merged to perform 10-fold cross validation. On each fold, a model was trained and the predictions were col-

[4] To enable replicability, the code is open sourced at https://github.com/SEERNET/EmoInt.

[5] https://www.github.com/felipebravom/AffectiveTweets

[6] http://www.cs.cmu.edu/~ark/TweetNLP/

lected on the remaining dataset. The predictions are averaged across all the folds to generalize the solution and prevent over-fitting. As described in Section 2.2, different combinations of feature extractors were used. After performing feature extraction, the data was then passed to various regressors Support Vector Regression, AdaBoost, RandomForestRegressor, and, BaggingRegressor of sklearn (Pedregosa et al., 2011). Finally, the chosen top performing models had the least error on evaluation metrics namely Pearson's Correlation Coefficient and Spearman's rank-order correlation.

2.4 Parameter Optimization

In order to find the optimal parameter values for the EmoInt system, an extensive grid search was performed through the scikit-Learn framework over all subsets of the training set (shuffled), using stratified 10-fold cross validation and optimizing the Pearson's Correlation score. Best cross-validation results were obtained using AdaBoost meta regressor with base regressor as XGBoost (Chen and Guestrin, 2016) with 1000 estimators and 0.1 learning rate. Experiments and analysis of results are presented in the next section.

3 Results and Analysis

3.1 Experimental Results

As described in Section 2.2 various syntax features were used namely, Part-of-Speech tags, brown clusters of TweetNLP project. However, these didn't perform well in cross validation. Hence, they were dropped from the final system. While performing grid-search as mentioned in Section 2.4, keeping all the lexicon based features same, choice of combination of emoji vector and word vectors are varied to minimize cross validation metric. Table 1 describes the results for experiments conducted with different combinations of word vectors. Emoji embeddings (Eisner et al., 2016) give better results than using plain GloVe and Edinburgh embeddings. Edinburgh embeddings outperform GloVe embeddings in **Joy** and **Sadness** category but lag behind in **Anger** and **Fear** category. The official submission comprised of the top-performing model for each emotion category. This system ranked 3rd for the entire test dataset and 2nd for the subset of the test data formed by taking every instance with a gold emotion intensity score greater than or equal to 0.5. Post competition, experiments were performed on ensembling diverse models for improving the accuracy. An ensemble obtained by averaging the results of the top 2 performing models outperforms all the individual models.

3.2 Feature Importance

The relative feature importance can be assessed by the relative depth of the feature used as a decision node in the tree. Features used at the top of the tree contribute to the final prediction decision of a larger fraction of the input samples. The expected fraction of the samples they contribute to can thus be used as an estimate of the relative importance of the features. By averaging the measure over several randomized trees, the variance of the estimate can be reduced and used as a measure of relative feature importance. In Figure 2 feature importance graphs are plotted for each emotion to infer which features are playing the major role in identifying emotional intensity in tweets. +/-EffectWordNet (Choi and Wiebe, 2014), NRC Hashtag Sentiment Lexicon, Sentiment140 Lexicon (Mohammad et al., 2013) and NRC Hashtag Emotion Lexicon (Mohammad and Kiritchenko, 2015) are playing the most important role.

3.3 System Limitations

It is important to understand how the model performs in different scenarios. Table 2 analyzes when the system performs the best and worst for each emotion. Since the features used are mostly lexicon based, the system has difficulties in capturing the overall sentiment and it leads to amplifying or vanishing intensity signals. For instance, in example 4 of fear **louder** and **shaking** lexicons imply fear but overall sentence doesn't imply fear. A similar pattern can be found in the 4th example of Anger and 3rd example of Joy. The system has difficulties in understanding of sarcastic tweets, for instance, in the 3rd tweet of Anger the user expressed anger but used **lol** which is used in a positive sense most of the times and hence the system did a bad job at predicting intensity. The system also fails in predicting sentences having deeper emotion and sentiment which humans can understand with a little context. For example, in sample 4 of sadness, the 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. The performance is

Emotion	**Systems**	**Pearsonr**	**Spearmanr**	**Pearsonr** ≥ 0.5	**Spearmanr** ≥ 0.5
Anger	Baseline	0.639583	0.628180	0.510361	0.475215
	Em0-Ed1-Gl0	0.659566	0.628835	0.536701	0.508762
	Em1-Ed1-Gl0	0.660568	0.631893	0.536244	0.511621
	Em0-Ed0-Gl1*	0.675864	0.656034	0.529404	0.512774
	Em1-Ed0-Gl1	0.678214	**0.658605**	0.527375	0.510436
	Ensemble	**0.678477**	0.653964	**0.540919**	**0.518851**
Fear	Baseline	0.631139	0.622047	0.476480	0.432407
	Em0-Ed1-Gl0	0.689571	0.66237	0.539250	0.499864
	Em1-Ed1-Gl0	0.695443	0.670438	0.542909	0.500896
	Em0-Ed0-Gl1	0.691143	0.667255	0.546867	0.510041
	Em1-Ed0-Gl1*	0.697630	0.676379	0.551465	0.510265
	Ensemble	**0.705260**	**0.683536**	0.**55641**	**0.513398**
Joy	Baseline	0.645597	0.652505	0.370499	0.363184
	Em0-Ed1-Gl0	0.696448	0.66237	0.539250	0.499864
	Em1-Ed1-Gl0	0.722115	0.720437	0.519821	0.508484
	Em0-Ed0-Gl1	0.689692	0.689883	0.472973	0.470260
	Em1-Ed0-Gl1*	0.714850	0.713558	**0.551191**	**0.543565**
	Ensemble	**0.728093**	**0.727970**	0.547213	0.537690
Sadness	Baseline	0.711998	0.711745	0.479049	0.452047
	Em0-Ed1-Gl0	0.737805	0.733999	0.547871	0.524843
	Em1-Ed1-Gl0*	0.744550	0.740893	**0.554723**	0.533571
	Em0-Ed0-Gl1	0.731436	0.724570	0.542910	0.536228
	Em1-Ed0-Gl1	0.736081	0.731050	0.553460	**0.548944**
	Ensemble	**0.748901**	**0.743589**	0.547213	0.537690
Average	Baseline	0.657079	0.653619	0.479049	0.452047
	Em0-Ed1-Gl0	0.695847	0.680207	0.51998	0.493755
	Em1-Ed1-Gl0	0.705669	0.690915	0.538424	0.513643
	Em0-Ed0-Gl1	0.69703	0.684436	0.523038	0.507326
	Em1-Ed0-Gl1	0.706694	0.694898	0.545873	0.528303
	Official*	0.708267	0.696801	0.546913	0.526018
	Ensemble	**0.715183**	**0.702265**	**0.55209**	**0.530501**

Table 1: Evaluation Metrics for various systems. Systems are abbreviated as following: For example `Em1-Ed0-Gl1` implies Emoji embeddings and GloVe embeddings are included, Edinburgh embeddings are not included in features keeping other features same. Results marked with * corresponds to official submission. Results in **bold** are the best results corresponding to that metric.

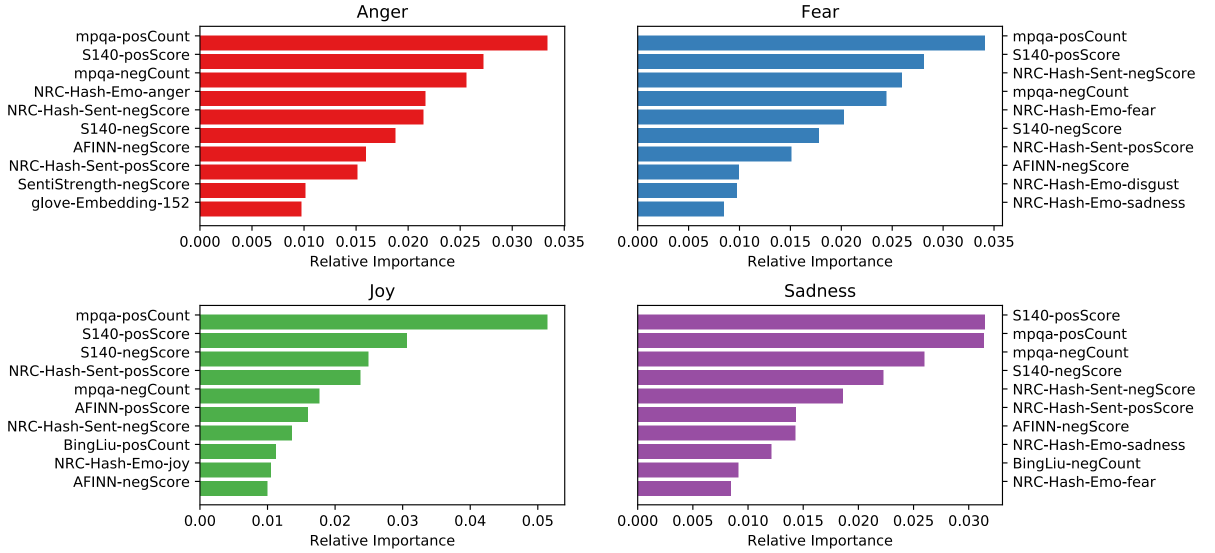

Figure 2: Relative Feature Importance of Various Emotions

poor with very short sentences as there are fewer indicators to provide a reasonable estimate.

4 Future Work & Conclusion

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 (EmoInt) to facilitate ease of experimenting with various lexicon features for text tasks is open-sourced. It provides plug and play access to various feature extractors and handy scripts for creating ensembles.

Few problems explained in the analysis section can be resolved with the help of sentence embeddings which take the context information into consideration. The features used in the system are generic enough to use them in other affective computing tasks on social media text, not just tweet data. Another interesting feature of lexicon-based systems is their 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.

Acknowledgement

We would like to thank the organizers of the WASSA-2017 Shared Task on Emotion Intensity, for providing the data, the guidelines and timely support.

References

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*. volume 10, pages 2200–2204.

Felipe Bravo-Marquez, Eibe Frank, Saif M Mohammad, and Bernhard Pfahringer. 2016. Determining word–emotion associations from tweets by multi-label classification. In *WI'16*. IEEE Computer Society, pages 536–539.

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.

Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics* 18(4):467–479.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, pages 785–794.

Yoonjung Choi and Janyce Wiebe. 2014. +/-effectwordnet: Sense-level lexicon acquisition for opinion inference. In *EMNLP*. pages 1181–1191.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bošnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359* .

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth*

Emotion	Tweet	Gold Int.	Pred. Int.
Anger	@Claymakerbigsi @toghar11 @scott_mulligan_ @BoxingFanatic_ Fucker blocked me 2 years ago over a question lol proper holds a grudge old Joe	0.625	0.6245
	We are raging angry.=1/2 bil $ for 2 pro Liars.(Actors) the most useless people in america Where is ours for working 100 X harder? @FoxNews	0.667	0.6665
	dammit @TMobile whays going on!!! 😤😤😤😤 lol #smh #mobilefails	0.792	0.4062
	People are #hurt and #angry and it's hard to know what to do with that #anger Remember, at the end of the day, we're all #humans #bekind	0.250	0.6040
Fear	Onus is on #Pak to act against #terror groups which find safe havens and all types of support for cross border terror: #MEA	0.667	0.6673
	Ffs dreadful defending	0.479	0.4795
	🎵 OLD FISH	0.070	0.5028
	@MannersAboveAll *laughs louder this time, shaking my head* That was really cheesy, wasn't it?	0.083	0.4936
Joy	@headfirst_dom I often imagine hoe our moon would feel meeting the jovial moons which are all special	0.500	0.5002
	Your attitude toward your struggles is equally as important as your actions to work through them.	0.340	0.3397
	Oi @THEWIGGYMESS you've absolutely fucking killed me.. 30 mins later im still crying with laughter.. Grindah.. Grindah... 🤓 hahahahahahaha	0.847	0.3726
	@WuffinArts :c You have my most heartfelt condolences. I'm glad it passed with levity and love in it's heart.	0.188	0.5872
Sadness	@nytimes media celebrated Don King endorsing #Obama in 08 and 12 now criticize him for endorsing #Trump who wants new Civil Rights era sad	0.562	0.5623
	@AFCGraMaChroi oh, sorry if I've discouraged you 😂	0.340	0.3397
	oh, btw - after a 6 month depression-free time I got a relapse now... superb #depression	0.917	0.462
	Ibiza blues hitting me hard already wow	0.833	0.4247

Table 2: Sample tweets where our system's prediction is best and worst.

ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pages 168–177.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Saif M Mohammad. 2017. Word affect intensities. *arXiv preprint arXiv:1704.08798* .

Saif M. Mohammad and Felipe Bravo-Marquez. 2017a. Emotion intensities in tweets .

Saif M. Mohammad and Felipe Bravo-Marquez. 2017b. Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*. Vancouver, Canada.

Saif M. Mohammad and Felipe Bravo-Marquez. 2017c. Wassa-2017 shared task on emotion intensity. EMNLP 2017 Workshop on Computational Approaches to Subjectivity, Sentiment, and Social Media (WASSA), Copenhagen, Denmark.

Saif M Mohammad and Svetlana Kiritchenko. 2015. Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence* 31(2):301–326.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242* .

Saif M Mohammad and Peter D Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*. Association for Computational Linguistics, pages 26–34.

Finn Årup Nielsen. 2011. A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903* .

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. volume 14, pages 1532–1543.

Sasa Petrovic, Miles Osborne, and Victor Lavrenko. 2010. The edinburgh twitter corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*. pages 25–26.

Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. 2010. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology* 61(12):2544–2558.