UWat-Emote at EmoInt-2017: Emotion Intensity Detection using Affect Clues, Sentiment Polarity and Word Embeddings

Vineet John
Cheriton School of Computer Science
University of Waterloo
vineet.john@uwaterloo.ca

Olga Vechtomova
Department of Management Sciences
University of Waterloo
ovechtom@uwaterloo.ca

Abstract

This paper describes the UWaterloo affect prediction system developed for EmoInt-2017. We delve into our feature selection approach for affect intensity, affect presence, sentiment intensity and sentiment presence lexica alongside pre-trained word embeddings, which are utilized to extract emotion intensity signals from tweets in an ensemble learning approach. The system employs emotion specific model training, and utilizes distinct models for each of the emotion corpora in isolation. Our system utilizes gradient boosted regression as the primary learning technique to predict the final emotion intensities.

1 Introduction

The goal of this EmoInt task is to predict the intensity of affect expressions in a selection of tweets. The intensity scores are floating point values between 0 and 1, representing low and high intensities of the emotion being expressed, respectively. The emotions analyzed in this shared task are anger, fear, joy and sadness (Mohammad and Bravo-Marquez, 2017b) (Mohammad and Bravo-Marquez, 2017a).

This paper describes the techniques used to clean tweets, build lexical features, find optimal combinations of features to produce a final vector representation of a tweet and train generalized regression, gradient boosted regression and neural-network computed regression models to fit the vector representations to the intensity scores.

The following sections describe each of these processes, followed by an enumeration of the parameters that worked in favor of the best-performing models, a discussion of the results and potential approaches to boost model accuracy.

2 Related Work

A majority of the existing literature on emotion/affect analysis on text focuses on classification tasks which aim to predict the probability distribution of a pre-defined set of emotions in bodies of text (Alm et al., 2005) (Aman and Szpakowicz, 2007) (Strapparava and Mihalcea, 2007). The VAD (valence, arousal and dominance) model as a way of visualizing multiple aspects of each known emotion was proposed by (Schlosberg, 1954), which has subsequently been adopted by other studies in quantifying emotion (Bradley and Lang, 1999).

This shared task is designed with the purpose of detecting intensity of a tweet given an emotion, which is comparable to detection of arousal to stimulus in the VAD model. The immediate difference that is noted compared to emotion classification tasks is that the training data can be annotated with cross-emotional intensity scores. The annotated scores for the tweets is obtained using Best-Worst Scaling, which increases the reliability of continuous valued scores (Kiritchenko and Mohammad, 2017).

3 Data Cleaning

Tweets, in general, are not always syntactically well-structured and the language used doesn't always strictly adhere to grammatical rules (Barbosa and Feng, 2010). Our feature extraction approach doesn't depend on syntactic features, relying solely on the presence of lexical features.

The grammatically incorrect use of language in many published tweets also makes it a necessity to clean the raw text in order to filter noisy data including special characters, alphanumeric strings, etc. The letter case for each tweet is standard-

ized by converting all tweets to lowercase. Stopwords are removed using NLTK (Bird, 2006). The hashtags in the tweets are stripped of the # symbol, and each of the hashtags are treated as regular unigrams in the corpus. The twitter handles are stripped away under the hypothesis that they are entity references that aren't correlated with affect.

All of the annotated lexica are also cleaned in the exact same way as the tweets are, to ensure that lexical pattern matching does not suffer as a result of the cleaning.

4 Feature Extraction

We used two primary methods for feature extraction from the tweets' raw text, namely annotated lexicons (Section 4.1) and pre-trained word embeddings (Section 4.2)

4.1 Annotated lexicons

Our system utilizes curated lexicons for emotion intensity/presence and sentiment intensity/presence. We include sentiment lexicons with the hypothesis that positive sentiment-polarity lexicon features would be positively correlated with some emotions and negatively correlated with others and vice-versa, since the emotion classes themselves possess an inherent sentiment polarity.

- **NRC Affect Intensity Lexicon (AI):** This lexicon assigns distinct emotion labels to unigrams, and provides the intensity at which the emotion is expressed. Each of the emotions evaluated in the EmoInt shared task are represented in this lexicon, and a floating point intensity score is assigned to each unigram-emotion pair (Mohammad, 2017).

- **NRC Emotion Lexicon (EL) & NRC Hashtag Emotion Lexicon (HE):** These lexicons contain the association of unigrams and Twitter hashtags with eight emotions (inclusive of the four emotions evaluated in this EmoInt task). EL is manually annotated on Amazon's Mechanical Turk (EL) and is scored either 0 or 1 implying whether or not the unigram is associated with any of the lexicon's eight emotion categories (Mohammad and Turney, 2010). HE is generated automatically from tweets with emotion-word hashtags and the features are floating point scores ranging from 0 to 2.24, indicating the intensity of the emotion category (Mohammad and Turney, 2013).

- **NRC Emoticon Lexicon (EC), NRC Hashtag Sentiment Lexicon (HS), NRC Emoticon Affirmative Context Lexicon and NRC Emoticon Negated Context Lexicon (EAN) & NRC Hashtag Affirmative Context Sentiment Lexicon and NRC Hashtag Negated Context Sentiment Lexicon (HSAN):** The first two lexicons associate words with positive/negative sentiment and the other two associate words with similar sentiment labels in affirmative or negated contexts generated automatically from tweets with sentiment-emoticons and sentiment-word hashtags. The terms in these lexicons can be unigrams, bigrams or pairs of unigrams and bigrams. The features are three-fold: a real-valued sentiment score denoted by the point-wise mutual information between a term and the positive/negative class, the number of times the term appears in each positive and negative contexts (Kiritchenko et al., 2014) (Mohammad et al., 2013) (Zhu et al., 2014).

- **SentiWordNet (SWN):** SentiWordNet is an opinion mining resource available through NLTK. Words in this lexicon are related in terms of synonymy. For each word present in the WordNet lexicon, three floating point sentiment scores are given: positive, negative and objective, such that

$$\sum_{i \in pos,neg,obj} word\_score_i = 1$$

  The positive and negative scores are extracted as features for each of the individual words present in the cleaned tweets. If a word does not have an entry or synonym in SentiWordNet, the positive and negative sentiment scores are assumed to be zero (Esuli and Sebastiani, 2007).

- **Emoji Valence (EV):** This is a hand-classified lexicon of Unicode emojis, rated on a scale of -5 (negative) to 5 (positive)[1].

- **Depeche Mood (DM):** This is a lexicon comprised of about 37,000 unigrams annotated with real-valued scores for the emotional states *afraid*, *amused*, *angry*, *annoyed*, *don't*

[1] https://github.com/wooorm/emoji-emotion

Emotion	Features	P	Sp	P (> 0.5)	Sp (> 0.5)
anger	W2V-GN, W2V-T, GV-T, AI, EL, EC, HS	0.705	0.686	0.521	0.507
fear	W2V-GN, W2V-T, GV-T, AI, SWN, EL, EC, EAN	0.713	0.694	0.558	0.525
joy	W2V-GN, GV-T, SWN, EC, HE, HS	0.728	0.705	0.619	0.599
sadness	W2V-T, GV-T, AI, SWN, EL, EC, EAN, HE, HS	0.679	0.668	0.507	0.468

Table 1: Training Cross-validated Accuracy

Emotion	Features	P	Sp	P (> 0.5)	Sp (> 0.5)
anger	W2V-GN, W2V-T, GV-T, AI, EC, HSL, GV-CC1, GV-CC2	0.691	0.670	0.581	0.556
fear	W2V-GN, W2V-T, GV-T, AI, SWN, EL, EC, EAN, HE, GV-WG, GV-CC2, EV	0.716	0.696	0.558	0.523
joy	W2V-GN, GV-T, AI, EC, HSL, HSAN, GV-WG, GV-CC1, EV	0.728	0.733	0.567	0.556
sadness	W2V-GN, W2V-T, GV-T, AI, SWN, EAN, HE, HSAN, GV-CC2, EV	0.729	0.723	0.550	0.535

Table 2: Testing Accuracy - Features + ML

care, *happy*, *inspired* and *sad* (Staiano and Guerini, 2014).

4.2 Word Embeddings

In addition to the features extracted from annotated lexica, vector representations of each of the tweets are generated from pre-trained word embeddings using large corpora. For our system, we utilize six distinct word embedding sources including two Word2Vec models, and four GloVe models.

- **Word2Vec Model - Google News (W2V-GN), Tweets (W2V-T):** Word2Vec is a technique for learning low-dimensional word embeddings for words in a corpus, based on the continuous bag-of-words (CBOW) and skip-gram models (Mikolov et al., 2013). W2V-GN is trained on the Google News corpus containing over 100 billion words. It is a skip-gram model containing 300-dimensional embeddings for 3 million distinct words and phrases[2]. W2V-T is a similar skip-gram model trained on tweets (Godin et al., 2015) and the embeddings produced are 400-dimensional and real-valued[3].

- **GloVe Model - Tweets (GV-T), Wikipedia + Gigaword (GV-WG), Common Crawl 42B tokens (GV-CC1), Common Crawl 840B tokens (GV-CC2):** GloVe is similar to Word2Vec, in that it obtains dense vector representations of words. GloVe builds a word co-occurrence matrix for the entire corpus prior to training. This matrix is then utilized to produce word and phrase vectors based on their context of appearance in the corpus (Pennington et al., 2014). The embeddings used in the system are 200- to 300-dimensional and real-valued[4].

The tweet vector representations using each of these word embeddings could be obtained either by averaging or summing up the real-valued word vectors for each of the words that had a corresponding trained vector representation from the pre-trained embeddings. Our system averages the word vectors, to avoid introducing a tweet length bias.

[2] https://code.google.com/archive/p/word2vec/

[3] http://www.fredericgodin.com/software

[4] https://nlp.stanford.edu/projects/glove

Emotion	P	Sp	P (> 0.5)	Sp (> 0.5)
anger	0.692	0.678	0.529	0.519
fear	0.713	0.701	0.553	0.531
joy	0.676	0.680	0.422	0.423
sadness	0.704	0.711	0.556	0.554

Table 3: Testing Accuracy: Pre-trained Embedding Features + Shallow Neural Network

5 Model Learning

Since the task requires the computation of a real-valued emotion intensity score for the tweets in the test set, we explored several regression methods.

The models initially tested including simple linear regression and generalized linear models like Gaussian process regression and Bayesian ridge regression.

We also conducted experiments using two feed-forward neural network (NN) architectures implemented in Keras[5]. The shallow NN architecture (Fig.1) uses a hidden layer densely connected to a sigmoid output neuron, while the deep NN architecture (Fig.2) uses iteratively smaller dense hidden layers culminating in a sigmoid output neuron.

The first layer for the shallow NN as well as all layers for the deep NN were comprised of densely connected ReLU activation units. The learning method used is stochastic gradient descent (SGD).

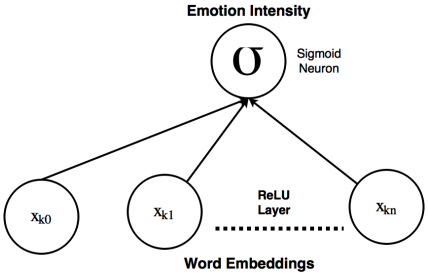

Figure 1: Shallow NN Architecture

However, all of these models were outperformed by gradient boosted regression models. The final system implementation uses the boosted regression implementation provided by the XGBoost library[6] (Chen and Guestrin, 2016).

6 System Tuning

The system was tuned with respect to feature selection by performing an exhaustive grid search

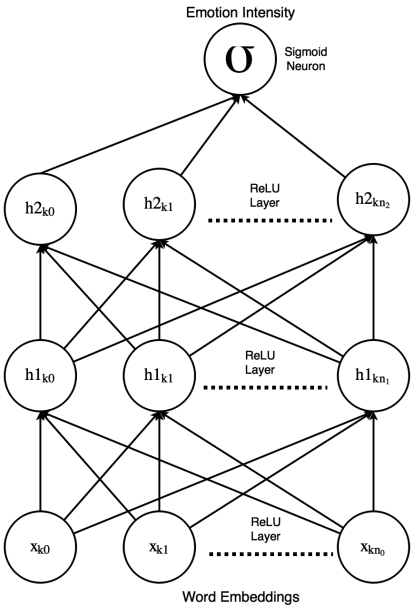

Figure 2: Deep NN Architecture

in the space of different possible combinations for the features. Consequently, the emotion intensity scores for each of the four emotions' test sets are predicted using models that have been trained on different subsets of the features, the accuracy results of which are discussed in Section 7.

Polynomial transformations of the features extracted from the annotated lexicons described in Section 4.1 were used to introduce non-linearity into the final feature space. The hyper-parameters of the gradient boosted regression model, namely tree-depth and number of boosted trees[7], were tuned using a randomized search strategy. The tree-depth retained it's library-default value of 3, and the number of boosted trees was set to 30,000.

Each of the feature sets was determined using 10-fold cross-validated evaluation on the combination of the training and development datasets.

[5] https://github.com/fchollet/keras

[6] http://dmlc.cs.washington.edu/xgboost.html

[7] http://xgboost.readthedocs.io/en/latest/python/python_api.html

7 Results

The systems in this shared task are evaluated using the Pearson correlation coefficient, which computes a bivariate linear correlation, and the Spearman rank correlation coefficient, which is a non-parametric version of the Pearson correlation coefficient, and relies on rank/ordering rather than absolute values (Mohammad and Bravo-Marquez, 2017b). These scores are denoted by **P** and **Sp**, respectively, in the results tables.

We present the results of the system submitted to the competition leaderboard in Table 1. The average scores of the system were 0.685 (Pearson) and 0.671 (Spearman). Post-competition evaluation on the gold labels of the test set are presented in tables 2 and 3. The correlation scores improved to 0.716 (Pearson) and 0.705 (Spearman) after grid-search testing including new features (EV & DM) using gradient boosted regression, as shown in table 2. Table 3 presents accuracy scores obtained using the Shallow NN architecture using only word embeddings as features.

Our system ranked 4^{th} overall, and 3^{rd} for the intensity range 0.5 to 1, on the task leaderboard.

8 Discussion

The results demonstrate that there is a different set of features that works best for each emotion in the task. It is observed that pre-trained word embeddings learned using Word2Vec and GloVe dominate the set of best performing features for nearly every emotion.

From experimental observations on the NN architectures in Keras, it was determined that increasing the depth of the network did not significantly improve its prediction accuracy. It was also noticed that the inclusion of regular & polynomial versions of the annotated lexicon features as features severely hampered the network's predictive accuracy. This could potentially be addressed by scaling each feature's values into a standard Gaussian distribution, or by clamping gradients to predetermined boundary values.

It is also worth noting that sentiment polarity lexicons boosted predictive accuracy for all four models, corroborating our hypothesis to justify their inclusion in the feature set.

9 Conclusion

We have described UWat-Emote, used at EmoInt to predict the emotion intensity of tweets. Our best system utilizes a combination of lexical resources and word embeddings to obtain vector representations of tweets, and uses gradient boosted regression to predict real-valued emotion intensities.

The system utilizes separate models for each emotion and achieves average Pearson and Spearman correlation scores of 0.716 and 0.705 respectively. Our implementation is fully open-sourced for replicability[8].

In the future, we would like to explore aspect based affect intensity for larger bodies of text, such as customer reviews for products and services. We would also like to evaluate normalized polynomial-kernel features and integrate the annotated lexicon features into convolutional and recurrent neural-network architectures.

Acknowledgments

We would like to acknowledge the organizers of this shared task, Saif M. Mohammad and Felipe Bravo-Marquez for their support.

We would also like to thank Saif M. Mohammad and Pierre Charron for permitting access to the NRC emotion and sentiment lexicons for this task.

References

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of the Conference on HLT/EMNLP*. Association for Computational Linguistics, HLT '05.

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *Text, speech and dialogue*. Springer, pages 196–205.

Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd CICLing: Posters*. Association for Computational Linguistics, pages 36–44.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*. Association for Computational Linguistics, pages 69–72.

Margaret M Bradley and Peter J Lang. 1999. Affective norms for english words (anew): Instruction manual and affective ratings. Technical report, Technical report C-1, the center for research in psychophysiology, University of Florida.

[8] https://github.com/v1n337/wassa-emoint-2017

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.

Andrea Esuli and Fabrizio Sebastiani. 2007. Sentiwordnet: A high-coverage lexical resource for opinion mining. *Evaluation* pages 1–26.

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. Multimedia lab@ acl w-nut ner shared task: named entity recognition for twitter microposts using distributed word representations. *ACL-IJCNLP* 2015:146–153.

Svetlana Kiritchenko and Saif M Mohammad. 2017. Best–worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of The Annual Meeting of the Association for Computational Linguistics (ACL). Vancouver, Canada*.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M Mohammad. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research* 50:723–762.

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. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*. Vancouver, Canada.

Saif M Mohammad and Felipe Bravo-Marquez. 2017b. Wassa-2017 shared task on emotion intensity. In *In Proceedings of the EMNLP 2017 Workshop on Computational Approaches to Subjectivity, Sentiment, and Social Media (WASSA)*. pages 26–34.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *In Proceedings of the seventh International Workshop on Semantic Evaluation Exercises (SemEval-2013)* page 321.

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.

Saif M Mohammad and Peter D Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence* 29(3):436–465.

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.

Harold Schlosberg. 1954. Three dimensions of emotion. *Psychological review* 61(2):81.

Jacopo Staiano and Marco Guerini. 2014. Depechemood: A lexicon for emotion analysis from crowd-annotated news. *arXiv preprint arXiv:1405.1605* .

Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*. Association for Computational Linguistics, pages 70–74.

Xiaodan Zhu, Svetlana Kiritchenko, and Saif M Mohammad. 2014. Nrc-canada-2014: Recent improvements in the sentiment analysis of tweets. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*. Citeseer, pages 443–447.