NUIG at EmoInt-2017: BiLSTM and SVR Ensemble to Detect Emotion Intensity

Vladimir Andryushechkin, Ian David Wood and James O' Neill
Insight Centre for Data Analytics,
National University of Ireland, Galway
{vladimir.andryushechkin,ian.wood,james.oneill}@insight-centre.org

Abstract

This paper describes the entry NUIG in the WASSA 2017[1] shared task on emotion recognition. The NUIG system used an SVR (SVM regression) and BiLSTM ensemble, utilizing primarily n-grams (for SVR features) and tweet word embeddings (for BiLSTM features). Experiments were carried out on several other candidate features, some of which were added to the SVR model. Parameter selection for the SVR model was run as a grid search whilst parameters for the BiLSTM model were selected through a non-exhaustive ad-hoc search.

1 Introduction

The WASSA 2017 shared task on emotion intensity (EmoInt) is a competition intended to stimulate research into emotion recognition from text (Mohammad and Bravo-Marquez, 2017). The task provides a corpus of 3960 English language tweets annotated with a continuous intensity score for each of four basic emotions: anger, fear, joy and sadness. This is a subset of the set of basic emotions proposed by Ekman (Ekman, 1992), which has been widely used as an emotion representation scheme in emotion recognition research (Mohammad, 2016; Poria et al., 2017). An additional 3142 tweets were used for evaluation of competition entries, with annotations withheld during the competition.

The NUIG entry to the task consisted of an ensemble of two supervised models: an SVR (Support Vector Machine Regression[2]) with n-gram and several custom features and a BiLSTM (Bidirectional Long-Short Term Memory[3]) model utilising tweet word embeddings. The models are accessible on DockerHub, GitHub and as a Rest API service (see Section 6).

In Section 2 we briefly overview related work. In Section 3 we discuss the data cleaning and preprocessing steps taken. In Section 4 we describe the model architectures and parameter choices. In Section 5 we discuss some observed issues with the models.

2 Related Research

In this section we briefly describe related work that has attempted to model emotions using machine learning based regressors and classifiers.

Wu et al. (Wu et al., 2006) use a hybrid of keyword search and Artificial Neural Networks (when no emotional keywords are present) to tackle the problem of detecting multiple emotions (anger, fear, hope, sadness, happiness, love and thank) achieving an average test accuracy for all emotions of 57.75 %. In the speech recognition domain, Wllmer et al. (Wöllmer et al., 2008) have applied Long Short Memory Networks (LSTMs) to detect emotions from speech using spectral features and measurements of voice quality, in an attempt to continuously represent emotions as opposed to using discrete classes of valence, arousal and dominance. Schuller et al. (Schuller et al., 2008) in 2008 combined both acoustic models of speech, phonetics and word features on the EMO-DB database[4] which demonstrated the importance of incorporating word models for such emotion recognition tasks.

[1] 8th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis

[2] http://scikit-learn.org/

[3] keras+theano: https://keras.io/

[4] see here: http://emodb.bilderbar.info/

Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pages 175–179
Copenhagen, Denmark, September 7–11, 2017.

3 Preprocessing

Tokenisation for both models was based on the regular expressions and rules provided with Stanford's Glove Twitter Word Vectors (Pennington et al., 2014) with some custom additions and modifications. Notable changes included the removal of hash symbols from tags, and extra emoticon detection patterns.

Removal of hash symbols had noticeable impact on the training accuracy for the BiLSTM model (for SVR it did not have significant impact). One possible explanation is the presence of hash tags in the training data for which the corresponding word is present in the word embedding, but not the tag itself. A concrete example is "#firbromyalgia". Note that stop words were not removed.

The preprocessing steps were as follows:

1. URL's, @mentions are replaced by standard tokens: "<url>" and "<user>"
2. emoticons were replaced by a small set of standard tokens: "<smile>", "<lolface>", "<sadface>", "<neutralface>", "<heart>"
3. hash symbols are removed from #hashtags
4. repeated full stops, question marks and exclamation marks are replaced with a single instance with a special token "<repeat>" added
5. characters repeated 3 times or more are replaced with one instance and a special token "<elong>" is added
6. a special token "<allcaps>" is added for each word in all capitals
7. remaining punctuation characters are treated as individual tokens
8. apostrophes are removed from negative contractions (e.g. "don't" is changed to "dont")[5]
9. other contractions are split into two tokens (e.g.: "it's" is changed to "it" and "'s")
10. tokens are converted to lower case

4 Model Architecture and Training

The overall model is a simple ensemble of an Support Vector Regression (SVR — see Section 4.1) and Bidirectional Long-Short Term Memory neural network (BiLSTM — see Section 4.2). The ensemble is described in Section 4.3.

The BiLSTM model was chosen due to it's recent excellent performance across numerous NLP tasks. The SVR model chosen as a baseline implementation, but found to contribute to the overall performance. Standard Long-Short Term Memory (LSTM) models were also attempted, however were outperformed by our BiLSTM (results not reported here).

[5]This transformation was evident from analysis of the word embedding dictionary

Emotion	C	gamma	epsilon	tol
anger	1.0	0.01	0.001	1e-04
fear	1.0	0.01	0.001	1e-04
joy	1.0	0.01	0.001	1e-05
sadness	1.0	0.001	0.001	1e-05

Table 1: Parameters for SVR models

4.1 Support Vector Machine Regression

The core features for the SVR model are a bag of 1,2,3 and 4-grams. N-grams with corpus frequency less than 2 or document frequency greater than 100 were removed. Experiments including words with document frequency up to 1000 showed similar performance, so the more stringent criterion resulting in a much smaller vocabulary was chosen. Note that this will also remove most words commonly considered stop words.

The following extra features were added. Average, min and max word vectors for each token are taken as features due to variation in sentence length[6]. Proportion of Capital symbols and proportion of words with first capital are considered. Finally, average, standard deviation, min and max of cosine similarities between the vector for each emotion name (e.g. "fear") and word vectors of all words in a tweet are added to the experiment.

An RBF (Radial Basis Function) kernel was chosen in preference to a Linear kernel as the classifier's training time is prompt due to the small dataset size. This kernel provided marginally better results.

A grid search of model parameters C, gamma, tolerance and epsilon was applied to find the optimal set parameters. The best combination is stored for each emotion model separately (see Table 1). Other model parameters were left at their default values in the `sklearn.svm.SVR` implementation as those values performed better than alternatives.

4.2 Bidirectional LSTM

Preprocessed and tokenized sentences are converted to 100-dimensional twitter Glove word vectors. We considered also 200–dimensional vectors[7], however performance was slightly worse and memory requirements substantially increased.

Embedding vectors were fed into a BiLSTM network followed by a layer trained with dropout (Srivastava et al., 2014) to reduce over-

[6]Length calculated before removing rare words/n-grams

[7]100d and 200d Glove Twitter 27B word vectors

fitting issues. The output of the dropout layer was inputted to a 2-hidden layer network before a final activation layer.Experiments were carried out on the 2-hidden layers where the number of neurons were varied between 20–60 in the first hidden layer and in the range of 10–20 in the second layer. For the sake of brevity, we only focus on the best performing architecture which is 100–50–25–1 (See Figure 1). Smaller layer sizes are not sufficient to catch the shape of the data and excessively big layer sizes lead to over-fitting and exponential growth of training time.

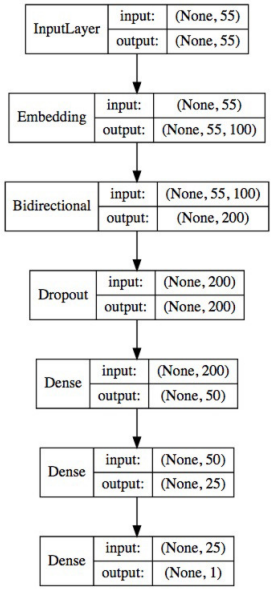

Figure 1: BiLSTM model architecture

For the loss function in training, Mean Absolute Error (MAE) is used in preference to Mean Squared Error (MSE) as it assigns equal weight to the data points and thus emphasizes the extremes. The "Softsign" activation function is found the best for the problem. Spearman and Pearson correlations are used as the main evaluation of network structures and parameter settings, however we also considered R2 scores, as in some cases Spearman and Pearson scores remained the same over training epochs while the R2 score improved.o

To avoid over-fitting, the number of training epochs is chosen through evaluating models after each epoch. The number of epochs at which training did not significantly improve Spearman correlation ρ is chosen for the final model (see Table 2). It is evident that fear takes considerably longer to train, 4 times longer than joy for example.

Emotion	anger	joy	fear	sadness
Training Epochs	12	8	36	18

Table 2: Number of BiLSTM training epochs.

Emotion	Estimator	R2	Pearson	Spearman
anger	svr	0.34	0.60	0.57
	lstm	0.36	0.63	0.61
	averaged	**0.42**	**0.66**	**0.63**
fear	svr	0.44	0.67	0.63
	lstm	0.45	0.68	0.66
	averaged	**0.49**	**0.71**	**0.68**
joy	svr	0.36	0.62	0.63
	lstm	0.35	0.59	0.59
	averaged	**0.41**	**0.65**	**0.65**
sadness	svr	0.43	0.68	0.69
	lstm	0.45	0.70	0.69
	averaged	**0.49**	**0.73**	**0.72**
average	averaged	**0.45**	**0.68**	**0.67**

Table 3: Performance comparison of individual and ensemble models evaluated on the WASSA test set.

4.3 Ensemble

With the limited time available, we attempted three simple approaches: taking the maximum, minimum and average of the predicted intensity between the two models. The best performance was obtained by averaging the LSTM and SVR outputs (see Table 3).

We believe that further investigation of the characteristics that led to a better ensemble model would likely lead to improvements in model design both in the BiLSTM itself and in alternative ensemble strategies.

5 Discussion

Overall, we see that performance in the development data set, used to select model parameters, did not differ substantially from performance on the test set, indicating that overfitting did not occur (see Table 4). Interestingly the difference between development and test set performance varies in line with the number of epochs. Concretely, *fear* and especially *sadness* see a strong performance gain on the test set, whereas the *joy* model degraded in performance, which was trained for the lowest number of epochs for all emotions. Given that our performance relative to the best performing entry also followed this pattern and that a dropout layer was used, which has been shown to control overfitting (Srivastava et al., 2014), it seems likely that choosing a larger number of epochs would have resulted in better models.

Analysis of model prediction errors on test data

Emotion	eval data	R2	Pearson	Spearman
anger	dev	**0.50**	**0.71**	**0.67**
	test	0.42	0.66	0.63
fear	dev	0.45	0.62	0.65
	test	**0.49**	**0.71**	**0.68**
joy	dev	**0.53**	**0.73**	**0.73**
	test	0.41	0.65	0.65
sadness	dev	0.26	0.52	0.56
	test	**0.49**	**0.73**	**0.72**
average	dev	0.43	0.64	0.65
	test	**0.45**	**0.68**	**0.67**

Table 4: Performance comparison between development and test sets.

revealed that extreme values were not modelled well for both SVR and BiLSTM models, with the SVR model performing marginally better, as seen for *anger* in Figure 2 (other emotions were similar). In the case of the BiLSTM model, we attribute this to the choice of L1 error as the loss function, which does not penalise large errors strongly. Overall performance with this loss function was, however, better on the provided data.

We also attempted to use the Emotion Hashtag Corpus (Mohammad, 2012) as training data for the BiLSTM model. This corpus only has category labels, so a model was built providing class probabilities, which were used as a proxy for intensity of the emotion classes. The performance was worse than random however, with an average R2 score of -3.63 (correlation 0.28), and this approach was abandoned. We believe this is due to two main factors: the intrinsic noise associated with emotion hash tags as emotion labels and that emotion probability is not a good analogue for emotion intensity. It would be interesting to experiment in the future with adding a binary feature for each emotion provided by a model trained on the hashtag corpus to our models.

6 Conclusion

The English language datasets provided for the WASSA competition are relatively clean but small, and the annotated labels for four emotions are precise and valuable. We performed experiments on the provided data drawing on our experience in emotion detection. The best built models are developed further and put together as an accessible service / software. The service is now available as part of the MixedEmotions platform[8] as well as the DockerHub as a docker image, on

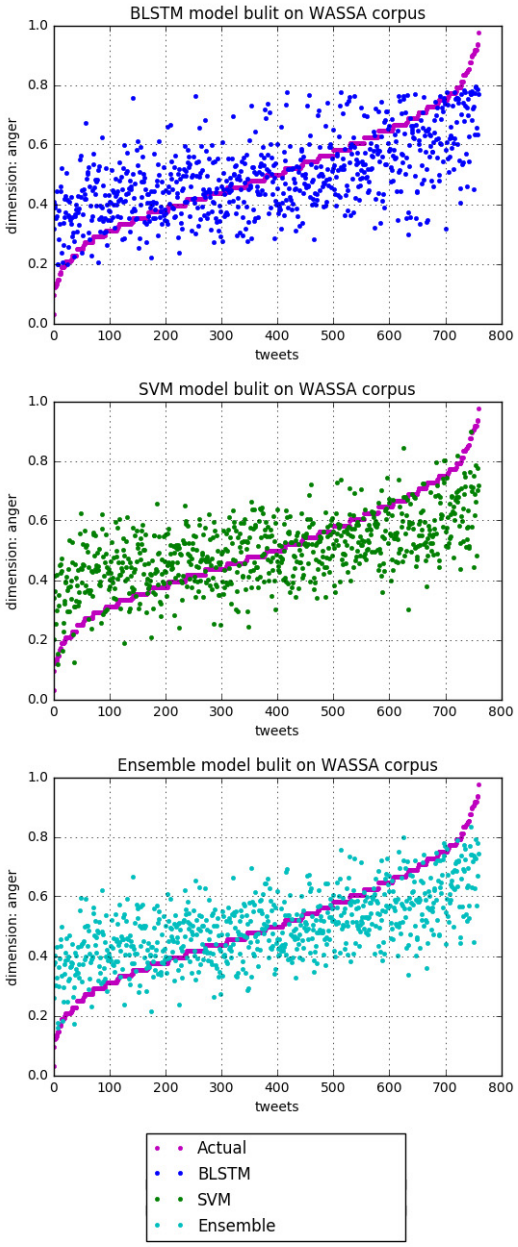

Figure 2: Model Predictions for *anger*. Other emotions follow a similar pattern.

[8]http://mixedemotions.insight-centre.org/

GitHub[9] DockerHub[10].

Acknowledgments

This work was supported in part by the European Union supported project MixedEmotions (H2020-644632) and the Science Foundation Ireland under Grant Number SFI/12/RC/2289 (Insight).

References

Paul Ekman. 1992. An argument for basic emotions. *Cognition and Emotion* 6(3-4):169–200.

Saif M. Mohammad. 2012. #emotional tweets. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, Stroudsburg, PA, USA, SemEval '12, pages 246–255.

Saif M. Mohammad. 2016. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In Herbert L. Meiselman, editor, *Emotion Measurement*. Woodhead Publishing, pages 201–237.

Saif M. Mohammad and Felipe Bravo-Marquez. 2017. WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*. Copenhagen, Denmark.

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.

Soujanya Poria, Erik Cambria, Rajiv Bajpai, and Amir Hussain. 2017. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion* 37:98–125.

Bjorn Schuller, Bogdan Vlasenko, Dejan Arsic, Gerhard Rigoll, and Andreas Wendemuth. 2008. Combining speech recognition and acoustic word emotion models for robust text-independent emotion recognition. In *Multimedia and Expo, 2008 IEEE International Conference on*. IEEE, pages 1333–1336.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15(1):1929–1958.

Martin Wöllmer, Florian Eyben, Stephan Reiter, Björn Schuller, Cate Cox, Ellen Douglas-Cowie, and Roddy Cowie. 2008. Abandoning emotion classes-towards continuous emotion recognition with modelling of long-range dependencies. In *Ninth Annual Conference of the International Speech Communication Association*.

Chung-Hsien Wu, Ze-Jing Chuang, and Yu-Chung Lin. 2006. Emotion recognition from text using semantic labels and separable mixture models. *ACM transactions on Asian language information processing (TALIP)* 5(2):165–183.

[9] https://github.com/MixedEmotions/05_emotion_wassa_nuig

[10] https://hub.docker.com/r/mixedemotions/05_emotion_wassa_nuig/