Deliverable 2

Sofia Y. Ahmed
syahmed@uw.edu

Libbey Brown
egollhof@uw.edu

Rachel Hantz
hantz1rk@uw.edu

Elizabeth Okada
esokada@uw.edu

Abstract

This project report provides information about the initial systems we have created for our primary task. Our primary task is multi-class emotion classification in English essays. Our adaptation task will be multi-class emotion classification in Urdu tweets.

1 Introduction

The primary task for our system this quarter is emotion classification in English essays. This task was Track 2 of the shared task for WASSA 2022 (Barriere et al., 2022) and required participants to develop a multi-class classification model which predicts an emotion tag given the text of an essay.

For this initial system, we experimented with six types of vector representations of words (baseline bag of words, lexical resources, and static embeddings) and two types of classifiers (decision tree (DT) and support vector machines (SVM)). WASSA 2022 defined the official competition metric for the task as the macro F1-score, so this is the metric we used to define the highest-performing system.

None of our systems yielded high performance on the classification task; the macro F1-score for our highest-performing system was only 24.36. This system was our baseline bag of words model with a decision tree classifier.

We believe the performance of our system can be improved by addressing several limitations. Moving forward, we intend to implement techniques to handle class imbalance, modify our pre-processing to take negation into account, use contextualized embeddings, and use boosting to reduce the errors made by our classifiers.

2 Task description

2.1 Primary Task

Our primary task is emotion classification for essay texts written in English; this task is Track 2 of the shared task for WASSA 2022 (Barriere et al., 2022). We will develop a multi-class classification model which predicts an emotion tag from one of seven categories: Ekman's six basic emotions (joy, sadness, surprise, disgust, anger, or fear) and the neutral tag no-emotion.

The dataset was initially generated by Buechel et al. (2018) and consists of 2655 essays which were written by 403 participants in reaction to reading disturbing news articles. The dataset was downloaded from the WASSA (2022) website. Table 1 describes the original train, development, and test splits.

Train	Dev	Test	Total
1860	270	525	2655

Table 1: Original Dataset Split for Primary Task

Due to the inavailability of the original test data, we opted to obtain to retain the original development data as is, but then select 10% of test data from the original training data. Table 2 describes the new train, development, and test splits. Here, there are 2130 total essays with a 77.3/12.7/10 train/dev/test split.

Train	Dev	Test	Total
1647	270	213	2130

Table 2: New Dataset Split for Primary Task

Participants rated their level of empathy and distress after reading the article and then described their thoughts and feelings in writing. Emotional tags were added to the essays as part of the WASSA 2021 shared task (Tafreshi et al., 2021). Emotion tags were added through prediction models (the specific models used were a gated RNN and a fine-tuned RoBERTA model) and the tags were manually verified by annotators from Amazon Mechanical Turk. Because these essays were generated in response to disturbing news articles, the distribution of emotion tags is not balanced; sadness is overwhelmingly the largest tag, followed by anger. See Table 3 for the distribution of the emotion tags.

	Train	Dev	Test	Total
joy	72	14	10	96
sadness	570	98	77	745
disgust	131	12	18	161
fear	173	31	21	225
anger	312	76	37	425
surprise	145	14	19	178
no-emo	244	25	31	300

Table 3: Emotion Distribution for Primary Task

For evaluation, we will be using the evaluation script that was provided by the task organizers on the website. We have simplified the evaluation script by removing code for the regression task tracks, which are not part of our project, and also by allowing for arbitrary paths and filenames. We have not modified the function that calculates the scores. Our system outputs a .tsv file with a single column of emotion predictions in string format. The evaluation file compares this output file with the gold standard file and performance is calculated through several measures: Macro F1-Score, Macro Recall, Macro Precision, Micro F1-Score, Micro Recall, Micro Precision, and Accuracy.

2.2 Adaptation Task

For our adaptation task, we plan to classify emotion for tweets in Urdu; this was Task A for the shared task at FIRE 2022 (Butt et al., 2023). As with our primary task, we plan to develop a multi-class classification model which predicts an emotion tag from one of seven categories: Ekman's six basic emotions (joy[1], sadness, surprise, disgust, anger, or fear) and the neutral tag no-emotion. The adaptation differs from our primary task in both language and genre.

The Urdu dataset consists of 9750 annotated tweets. The datatset was downloaded from the FIRE (2022) shared task website. Table 4 lists the split between training and test data. To mirror the primary task, we may create a development set extracted from the training data prior to model development.

Train	Test	Total
7800	1950	9750

Table 4: Dataset Split for Adaptation

The tweets are labeled with emotional tags among Ekman's six basic emotions and the neutral no-emotion tag. Construction of the dataset is detailed in Ashraf et al. (2022)[2]. The authors' goal was to construct a balanced dataset, so Twitter hashtags were used to identify tweets representative of a particular emotion, and then annotators classified the tweets with one or more emotion tags. Note that this dataset differs from the dataset used in our primary task in that each tweet can be tagged with more than one emotion (i.e. suggestive of a multi-label classification task). We will utilize only the tweets which have a single emotion tagged for ease of adaptation. Table 5 shows the distribution of emotion tags for the full Urdu data, prior to selecting only tweets with one emotion tag[3].

	Train	Test	Total
joy	1046	261	1307
sadness	1550	388	1938
disgust	761	190	951
fear	609	152	761
anger	811	203	1014
surprise	1550	388	1938
no-emo	3014	753	3767

Table 5: Emotion Distribution for Adaptation

[1]The original Urdu dataset uses 'happiness' in place of 'joy'. We map the label 'happiness' to 'joy.'

[2]The FIRE 2022 shared task references Ashraf et al. (2022) as a source for their dataset, but the present dataset appears to be a slightly larger sample than the one depicted in Ashraf et al. (2022). Thus, we display counts from the present dataset, not those described by Ashraf et al. (2022).

[3]285 train and 61 test tweets had no emotion tag.

Table 6 lists the split between training and test data after selecting only tweets with one emotion tag.

Train	Test	Total
5463	1389	6852

Table 6: Dataset Split for Adaptation: Only One Emotion Tag per Tweet

Table 7 shows the distribution of Urdu emotion tags after selecting only tweets with one emotion tag.

	Train	Test	Total
joy	744	185	929
sadness	911	230	1141
disgust	20	10	30
fear	159	41	200
anger	89	35	124
surprise	526	135	661
no-emo	3014	753	3767

Table 7: Emotion Distribution for Adaptation: Only One Emotion Tag per Tweet

We will use the same evaluation script as for the primary task.

3 System Overview

Our system consists of four components:

1. Pre-Processing
2. Embeddings
3. Classification
4. Evaluation

For this deliverable, we experimented with several types of vector representations and two different classifiers. Further details regarding text preprocessing, generation of embeddings, and classification models can be found in Section 4. For the evaluation step, we followed the methods used in the WASSA 2022 shared task (Barriere et al., 2022); this process is detailed in Section 2.1. A summarization of the methods we used in these experiments can be found in Figure 1.

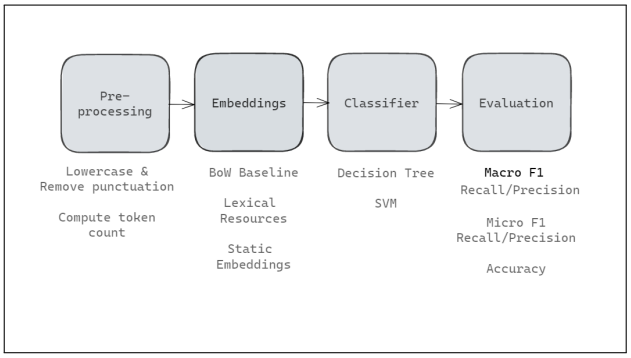

Figure 1: System Overview

The WASSA 2022 shared task defined the macro F1-score as the official competition metric, so this is the score we used to determine which of our systems had the best performance. We found that the system with the best macro F1-score was the baseline bag of words embeddings used with the Decision Tree classifier, closely followed by emotion lexicon enhanced bag of words embeddings and pre-trained Word2Vec static embeddings with the Decision Tree classifier.

4 Approach

4.1 Text Pre-processing

Minimal pre-processing was done for this system. We lowercased all characters and removed all punctuation and numbers. No stemming or stopword removal was done. There was no pre-processing done to handle negation.

4.2 Vector Representations

We used various methods to create feature vectors for the essays, including a bag of words baseline, lexical resources, and static embeddings. We describe these methods in the following sections.

4.2.1 Bag of Words

We utilized bag of words to use as a baseline vector representation for our system. We instantiated a vocabulary for the bag of words by sorting the set of tokens in all training instances combined in order to assign each token an index. Then, to vectorize an essay instance, we assigned the frequency of each token in that essay to that token's vocabulary index in the vector. If an essay instance contained a token not found in the vocabulary, it was interpreted as an "unknown" token, and assigned the index for "unknown" for smoothing.

4.2.2 Lexical Resources

We utilized the NRC Word-Emotion Association Lexicon (also known as EmoLex) as a resource to create two additional vector representations that go beyond a baseline bag of words: emotion only vectors and emotion enhanced bag of words vectors (Mohammad and Turney, 2010, 2013). For a set of 14,182 unigrams, EmoLex attributes a binary 0 (not associated) or 1 (associated) for eight emotions (anger, anticipation, disgust, fear, joy, sadness, surprise, trust) and two sentiments (negative and positive). This lexicon was initially and further developed using crowdsourcing.

Emotion only vectors contained 6 features, one for each of Ekman's 6 emotions. For each token in an essay instance that was also present in the emotion lexicon, we summed the binary feature vector returned for each word from EmoLex. If a token was present more than once in the essay instance, the corresponding binary feature vector was summed that many times. A vector containing all zeros, indicates that no tokens in the essay instance had any association with any of Ekman's 6 emotions according to EmoLex. Emotion enhanced bag of words vectors were the concatenation of the bag of words and emotion only vectors.

4.2.3 Static Embeddings

s We used three versions of static embeddings in our system. We used two versions of Word2Vec (one pre-trained, and one trained on a corpus of our essays) and pre-trained GloVe embeddings. The Python package Gensim was used to implement these embeddings [4].

Word2Vec is a model architecture which computes vector representations of words by reconstructing their linguistic context (Mikolov et al., 2013a). A large corpus of words is taken as an input and the model produces a vector space, generally of hundreds of dimensions, in which each word has a corresponding vector. These vector representations of words can be used as inputs in other machine learning tasks.

Two versions of Word2Vec vector representations were tested in our system. The first leveraged pretrained Word2Vec embeddings which were generated by training the model on a corpus of one billion words that came from a dataset of Google News articles (Mikolov et al., 2013b). The pretrained vector representations were obtained through Gensim's data repository (RaRe-Technologies). For the second version, a Word2Vec model was trained on our corpus of essays, and those vectors were used as the representations. We used Gemsim's default model settings when training our Word2Vec model.

GloVe is model that computes vector representations of words by determining how frequently words co-occur with each other (Pennington et al., 2014). As with Word2Vec, a large corpus of words is taken as an input to train the model, which produces vector representations of words. We utilized pre-trained GloVE vectors which were trained on a corpus of 27 billion tokens from two billion tweets; these embeddings were obtained through gensim's data repository (RaRe-Technologies).

We chose these particular pre-trained embeddings because they were pre-trained with data from different domains. As noted earlier, the Word2Vec pre-trained embeddings were trained on a corpus of Google News articles, and the GloVE pre-trained embeddings were trained on a corpus of Twitter data. The essays in our dataset are longer than tweets, but are written in a less formal style than a professionally published news article. We were interested in seeing if embeddings pre-trained on data from different domains would yield differences in performance on our classification task.

For all static embedding vectors, the representation of an essay - the centroid - was found in the same manner. First, the vector representation of each word was found; if there was no vector representation of a word it was omitted. The centroid of an essay was the average of the vector representations of all words. This method - calculating the centroid from the average of all words in the essay - was used in (Gennaro and Ash, 2022).

4.3 Machine learning models

Two Machine Learning models were used in our experimentation: Support Vector Machines (SVM) and Decision Trees. SVMs are supervised learning methods that can be used for classification, regression and detecting outliers (Pedregosa et al., 2011). In this case, we used SVM for classification. We first attempted to classify using SVM because it is commonly used in classification problems. Next, we implemented Decision Trees in order to experiment with a different Machine Learning model and see if we could improve the classification predictions from the experiments that used SVM. Deci-

[4] https://radimrehurek.com/gensim/index.html

sion Trees are also a supervised learning method used for classification and regression, and again, in our case, were used for classification. The Python package scikit-learn was used to implement both of these Machine Learning models. [5]

5 Results

When using SVM as the classifier, emotion prediction yields high macro precision. On average, when these models predict an essay to be a certain emotion, they are correct 73.3 - 90.9% of the time. However, macro recall is quite low. When an essay *should* be a certain emotion, these models are only 14.2 - 19.4% correct on average. Overall this indicates, that the models missclassify many instances, even if among individual predicted classes, there are a large proportion of true positives. Macro F1 score is exceedingly low, only ranging from 7.6 to 15.9. Emotion lexicon enhanced bag of words has the highest performance on average as seen in Table 8. Accuracy (micro F1/precision/recall) without respect to any class is higher at 36.3-42.2% correct predictions, but this measurement is misleadingly "high" due to our extreme class imbalance.

Vector	**Mac F1**	Mac R	Mac P	Mic F1
BoW	13.69	**17.53**	73.36	40
Emo	11.19	16.06	86.09	39.63
EmoBoW	**15.95**	19.42	74.95	**42.22**
W2V	7.61	14.29	**90.9**	36.3
W2VPT	7.61	14.29	**90.9**	36.3
GloVePT	7.61	14.29	**90.9**	36.3

Table 8: Macro F1-score, macro recall, macro precision, and micro F1-score using *SVM* classifier on the development set, where micro F1-score is equivalent to accuracy and micro recall and precision. Macro F1 is denoted in **bold** as it is the official shared task competition metric. BoW is a baseline system. The highest score for each metric is denoted in **bold**.

Using decision trees as the classifier yields much lower macro precision and slightly higher recall than SVM. As seen in Table 9, the bag of words baseline is the highest performer. On average, only 24.3% of predicted emotions are true positives and only 25.9% of emotions that *should* be predicted are detected. In general, these models struggle to detect emotions and predict emotions correctly. The macro F1 score (the official shared task competition metric) for the bag of words baseline using decision trees is our highest performer of all models at 24.3. A close second best performer is the emotion lexicon enhance bag of words model at 21.5. This vectorization schema was also a "high" performer with SVM. Nonetheless, these models will need to be iterated on in order to successfully move past a baseline as our best system.

Vector	**Mac F1**	Mac R	Mac P	Mic F1
BoW	**24.36**	**25.96**	**24.32**	**34.07**
Emo	19.93	22.12	20.03	25.93
EmoBoW	21.56	23.52	21.54	32.59
W2V	10.22	10.09	11.03	16.67
W2VPT	20.34	21.13	20.23	30.74
GloVePT	9.16	8.8	10.42	17.04

Table 9: Macro F1-score, macro recall, macro precision, and micro F1-score using *decision tree* classifier on the development set, where micro F1-score is equivalent to accuracy and micro recall and precision. Macro F1 is denoted in **bold** as it is the official shared task competition metric. BoW is a baseline system. The highest score for each metric is denoted in **bold**.

6 Discussion

Our current six embedding options in combination with our two classifier options yield low performance. It is likely that this is due to our class imbalance. The emotion *sadness* is the ground truth emotion for 35% of essays despite *sadness* being only 1 of 7 possible labels. We can clearly see the impact of this class imbalance in the predictions made by the SVM classifier. In the case of predictions made with static embeddings, all instances are predicted with the majority class. In Table 8, the metrics for each of these vectors are identical. For other embeddings, sadness is still the highest predicted emotion from a quick visual inspection. In the case of emotion lexicon only vectors predicted with SVM, we see just a few *anger* predictions–the second highest class. This explains the general trend of high precision (i.e many true positives for sadness), but low recall (i.e. many false negatives elsewhere) for SVM. It is clear that we will need to apply techniques to handle class imbalance.

When looking at predictions made using decision trees, we see a variety of emotions predicted, not just those limited to the majority classes. However, as made clear by the metrics displayed in Table 9, while these predictions varied, as the de-

[5] https://scikit-learn.org/stable/index.html

velopment set ground truth intended, they were ultimately incorrect. An ad-hoc confusion matrix of the baseline BoW system using decision trees is displayed in Figure 10. In an ideal system, all cells save for the diagonal would have a zero value. In our current best performing decision tree system, the true class is the highest prediction only 4/7 times. Meanwhile, there are numerous false positives and negatives. Often, the majority classes are predicted in the case of false predictions, once again. This again points to our need to handle class imbalance in our data.

	A	D	F	J	N	Sa	Su
A	24	9	3	4	14	15	7
D	3	5	1	0	1	1	1
F	3	3	9	2	6	7	1
J	6	0	2	1	1	3	1
N	6	1	3	2	2	8	3
Sa	13	7	10	4	12	49	3
Su	4	1	3	1	1	2	2

Table 10: Confusion matrix displaying the number of predictions made on the BoW baseline using decision trees on the development data. The vertical axis indicates the ground truth emotion in alphabetical order: Anger, Disgust, Fear, Joy, Neutral, Sadness, Surprise. The horizontal axis indicates the predicted emotion. The highest predicted emotion for each class is underlined.

It is interesting to us that the baseline bag of words model performed better than other techniques which leveraged lexical resources or static embeddings when using decision trees. This could have occurred for multiple reasons. EmoLex only takes into account emotional tags on single words. But, we examine words within unique contexts and used in sentences next to numerous other words; the emotionality of words may shift depending on their context. Moreover, in this slice, we have not handled the impact of negation. It is quite possible that we add emotional features opposite of the expressed emotional direction when creating vectors.

We further hypothesize that the poor performance of the GloVe embeddings may be due to the domain of the training data; possibly the corpus of Tweets was too dissimilar to our corpus of essays. The Word2Vec model we trained on our corpus of essays most likely yielded poorer performance because our corpus is quite small and so was insufficient training data.

7 Ethical Considerations and Limitations

7.1 Ethical Considerations

There are some ethical concerns regarding the collection of this dataset. These essays were written in response to disturbing news articles; as noted earlier, the most prevalent emotion in this dataset is sadness. It is likely that many of the participants had unpleasant reactions to the news articles and may have had emotional harm from participating in the creation of this dataset.

There is inherent risk in classifying human emotions based on text, especially if some significant decision will be made about the author based on the predicted emotion. One example is the detection of crisis in student standardized test essays (Burkhardt et al., 2021). There is significant doubt that machine learning algorithms are able to detect internal emotional states, and such systems also demonstrate racial and gender bias(Boyd and Andalibi, 2023). Added to this is the fact that our task only allows for one emotion to be detected per passage, which leaves little room for nuance in interpreting the results.

7.2 Limitations

There were several limitations to our experimentation for this deliverable. First was the fact that there was no available test data for this shared task, so our dataset, which was small to begin with, had to be further divided to use some of the existing data as a test dataset. Furthermore, the dataset was unbalanced in that a large portion of the training data was classified as sadness, which led to unbalanced predictions on the test dataset. This limited our ability to achieve higher accuracy scores between our predictions and the gold standard predictions. We hope to utilize weighted metrics during our next deliverable to help us gain a fuller perspective into the performance of our classifier.

During the preprocessing of the data, we did not handle negation, which was another limitation that affected the downstream process of prediction of emotion on the test data.

There were also some technological limitations. We were unable to leverage contextualized vectors such as ELMo because none of our machines were able to successfully run install and run the model, largely due to a lack of compute power.

Another limitation was that of the classifiers we used for this deliverable. While SVMs and Decision Trees are common classification models, they

are not neural classifier models, and therefore have limited ability to handle an unbalanced dataset and generate accurate predictions on test data.

All of these limitations will be used to inform future directions for the next deliverable.

8 Conclusion

In the development of this initial system, we experimented with different vector representations and classifiers. None of our systems yielded high performance on the classification task, and none of the systems performed better than the baseline bag of words model using the decision tree classifier. Overall, the SVM classifier overwhelmingly predicted the majority class, while the decision tree classifier was generally inaccurate.

Moving into the next deliverable, we hope to see improvements upon the performance of the baseline system by addressing many of the limitations in our current system. We plan to improve our text pre-processing to handle negation. We will utilize contextualized word embeddings. Additionally, we will experiment with techniques to handle class imbalance and weighted metrics for more granular insight during evaluation.

References

Noman Ashraf, Lal Khan, Sabur Butt, Hsien-Tsung Chang, Grigori Sidorov, and Alexander Gelbukh. 2022. Multi-label emotion classification of urdu tweets. *PeerJ Computer Science*, 8:e896.

Valentin Barriere, Shabnam Tafreshi, João Sedoc, and Sawsan Alqahtani. 2022. WASSA 2022 shared task: Predicting empathy, emotion and personality in reaction to news stories. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 214–227, Dublin, Ireland. Association for Computational Linguistics.

Karen L. Boyd and Nazanin Andalibi. 2023. Automated Emotion Recognition in the Workplace: How Proposed Technologies Reveal Potential Futures of Work. *Proceedings of the ACM on Human-Computer Interaction*, 7(CSCW1):95:1–95:37.

Sven Buechel, Anneke Buffone, Barry Slaff, Lyle Ungar, and João Sedoc. 2018. Modeling empathy and distress in reaction to news stories. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4758–4765, Brussels, Belgium. Association for Computational Linguistics.

Amy Burkhardt, Susan Lottridge, and Sherri Woolf. 2021. A Rubric for the Detection of Students in Crisis. *Educational Measurement: Issues and Practice*, 40(2):72–80.

Sabur Butt, Maaz Amjad, Fazlourrahman Balouchzahi, Noman Ashraf, Rajesh Sharma, Grigori Sidorov, and Alexander Gelbukh. 2023. Emothreat@fire2022: Shared track on emotions and threat detection in urdu. New York, NY, USA. Association for Computing Machinery.

FIRE. 2022. Emothreat at fire 2022.

Gloria Gennaro and Elliott Ash. 2022. Emotion and reason in political language. *The Economic Journal*, 132(643):1037–1059.

Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Saif Mohammad and Peter 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*, pages 26–34, Los Angeles, CA. Association for Computational Linguistics.

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

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 *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

RaRe-Technologies. gensim-data.

Shabnam Tafreshi, Orphee De Clercq, Valentin Barriere, Sven Buechel, João Sedoc, and Alexandra Balahur. 2021. WASSA 2021 shared task: Predicting empathy and emotion in reaction to news stories. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 92–104, Online. Association for Computational Linguistics.

WASSA. 2022. Wassa 2022 shared task.