Multi-Class Emotion Classification in English Essays and Urdu Tweets

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 final system we have developed for emotion classification in English and Urdu text. Our primary task is multi-class emotion classification in English essays. The best system is an ensemble of fine-tuned RoBERTa models and an emotion lexicon enhanced traditional decision tree approach. Our adaptation task is multi-class emotion classification in Urdu tweets. The best system is a fine-tuned RoBERTa model on an unbalanced dataset.

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.

Our adaptation task for the system is emotion classification in Urdu tweets. This task was Task A of EmoThreat at FIRE 2022 (FIRE, 2022), in which participants developed a multi-label classification system to predict one or more emotion tags given the text of a tweet. We modified this task to better align with our primary (making it a multi-class task) and so only utilized tweets from the task with a single emotion tag.

To develop our initial baseline on the primary task, we experimented with several combinations of embedding strategies and classification models. None of our systems yielded high performance on the classification task; the macro F1-score for our highest-performing baseline system was only 24.36 on the development data. This system combined a bag of words embedding strategy with a decision tree classifier.

We experimented with several modifications to the baseline system and found that our best-performing system utilized a pre-trained RoBERTa model fine-tuned on a version of our training data which was sampled to improve the balance of class labels. This system achieved a macro F1 score of 45.88 on the development data.

For our final system, we utilized ensembling to improve upon the performance of our system on the primary task, achieving a macro F1 score of 50.26 on the development data. We also adapted our system to work with the Urdu tweet data of our adaptation task. Our best system for the Urdu task was a pre-trained Urdu RoBERTa model which was fine-tuned on an unbalanced version of our training data, achieving a macro F1 score of 52.92 on the development data. Our final macro F1 scores on the test data were 36.97 for the primary task and 48.43 for the adaptation task.

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 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 indicating 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
neutral	244	25	31	300

Table 3: Emotion Distribution for Primary Task

For evaluation, we use 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 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 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 indicating 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 dataset distribution, we will create a development set extracted from the test 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

[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).

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
neutral	3014	753	3767

Table 5: Emotion Distribution for Adaptation

Table 6 lists the split between training, dev, and test data after selecting only tweets with one emotion tag, and then splitting the resulting test set (20%) into a development and test set. Here, there are a total of 6852 tweets with an approximate 80/10/10 train/dev/test split.

Train	Dev	Test	Total
5463	685	704	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	Dev	Test	Total
joy	744	89	96	929
sadness	911	108	122	1141
disgust	20	5	5	30
fear	159	21	20	200
anger	89	16	19	124
surprise	526	64	71	661
neutral	3014	382	371	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 three core approaches: (A) traditional machine learning classifiers, (B) finetuning a pre-trained language model, and (C) an ensemble voting method that selects best two-out-of-three results from the best three performing of these systems.

Our traditional machine learning classifier path (A) includes pre-processing, optional negation handling, multiple types of embeddings, optional class balancing, and a decision tree (DT) or support vector machine (SVM) classifier with optional boosting.

The pre-trained language model path (B) consists of optional class balancing and a pre-trained RoBERTa model that has been fine-tuned on the training data from either our primary task (English) or adaptation task (Urdu).

The ensemble voting method (C) selects best two-out-of-three results from our three best non-ensembled systems: RoBERTa with class balancing, RoBERTa without class balancing, and balanced emotion-enhanced BoW vectors with a decision tree classifier. This strategy was only implemented for our primary task.

4 Approach

4.1 Text Pre-processing

4.1.1 Primary Task - English

Text pre-processing for English included tokenization, lowercasing, removal of punctuation, and obtaining a vocabulary count of tokens.

We processed negated words using a technique adapted from the method laid out in (Babanejad et al., 2020), in which negation words were removed and the word immediately following the negation was replaced with its antomym. We used the Python package spaCy[4] to create dependency parses of each essay. Tokens with the dependency ”negation” were identified and removed; tokens that were the head of the negation tokens were replaced with their antonyms. Antonyms were identified using the Python package spacy-wordnet [5].

4.1.2 Adaptation Task - Urdu

Text pre-processing for Urdu included tokenizing, removal of numbers, removal of punctuation, and obtaining a vocabulary count of tokens.

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

[4] https://spacy.io/

[5] https://pypi.org/project/spacy-wordnet/

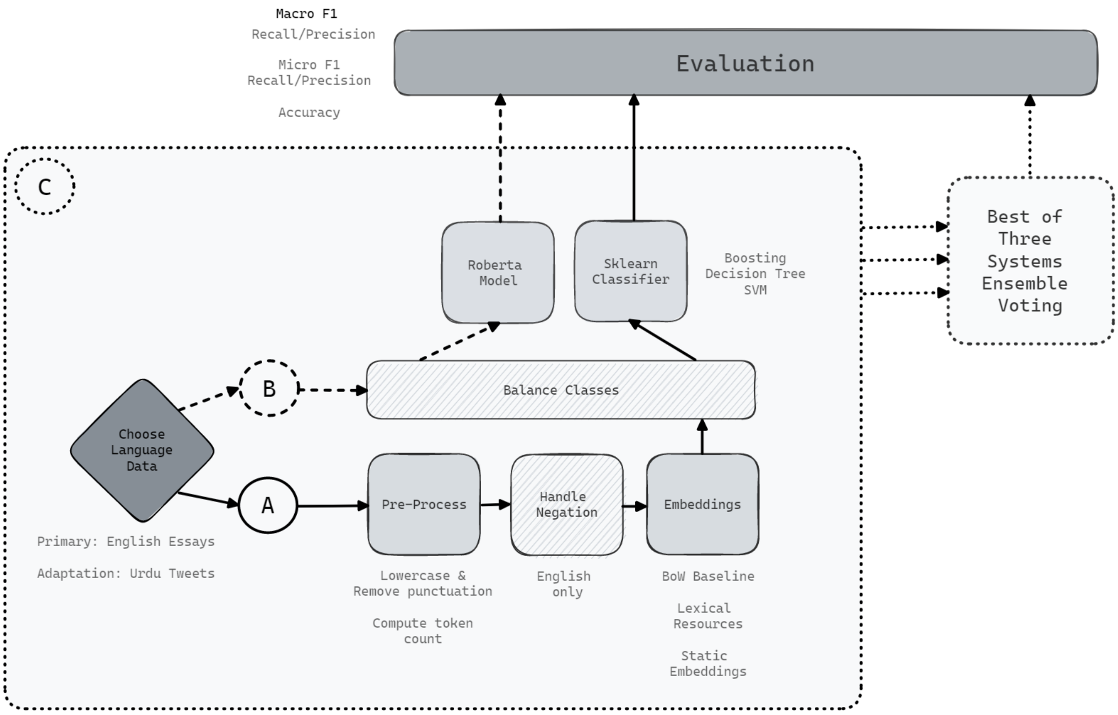

Figure 1: System Overview

4.2 Embeddings

We used various methods to create feature vectors for the essays, including a bag of words baseline, lexical resources, and static embeddings. We chose not to adapt the lower-performing embedding strategies to work for our Urdu adaptation task. We describe these methods in the following sections.

4.2.1 Bag of Words

This embedding strategy was used for both our primary English and adaptation Urdu tasks. 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

This embedding strategy was used for both our primary English and adaptation Urdu tasks. 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

This embedding strategy did not yield high performance on our primary English task, so it was not adapated for the Urdu system.

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. For this system, we utilized 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).

Each essay was represented with a centroid vector. 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 Class Imbalance

As seen in Tables 3 and 7, the distribution of emotions is highly skewed amongst the datasets of English essays (35% of essays are tagged as *sadness*, while only 4.5% of essays are tagged as *joy*) and Urdu tweets (55% of tweets are tagged as *no-emo*, while less than 1% of tweets are tagged as *disgust*). To handle class imbalance, we implemented and tested three class balancing strategies: random over/under sampling, SMOTE, and SMOTE with removal of Tomek links.

Random over/under sampling found the average class size for the training data, undersampling classes above the average class size and oversampling classes below the average class size so that all classes had the average class size. SMOTE (Synthetic Minority Over-sampling Technique) calculated intermediate vectors between pairs of data points of a minority class and added these synthetic minority class vectors to the training data (Chawla et al., 2002). Tomek links are pairs of opposite class instances that are each other's nearest neighbor. SMOTE with removal of Tomek links first applied the SMOTE algorithm and then undersampled the training dataset by removing any Tomek links (Batista et al., 2003). Both SMOTE strategies inserted synthetic vector instances up to the value of the majority class size. The python library imbalanced-learn was leveraged for all three methods[6].

We selected random over/under sampling as our class balancing strategy as it had the highest performance on our data. We describe our process for selecting this method in Section 5, displaying results of various trials in Table 9.

4.4 Classifiers

Our system used both our classical machine learning models (SVM and DT) and our RoBERTa models. For our primary task, we utilized an ensemble voting strategy, and for our adaptation task, we fine-tuned a new RoBERTa model to handle Urdu text.

Two classical machine learning models were used in our initial 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). 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. The Python package scikit-learn was used to implement both of these Machine Learning models. [7]

As an attempted improvement for the classification step, we implemented boosting. Boosting can be used in tandem with classical machine learning models, such as Decision Trees, to enhance the results. Therefore, we implemented boosting in tandem with both SVMs and Decision Trees, as those were the machine learning models that we used in our baseline system. Once again, scikit-learn was used for this.

For our pre-trained models path, our primary task (English) uses the *distilroberta-base* model from the Hugging Face hub. The transformers library was used to fine-tune two versions of this model for sequence classification on our English training data (unbalanced and balanced). We adopted the default hyperparameters recommended

[6] https://imbalanced-learn.org/stable/index.html

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

on the Hugging Face website.

For the adaptation task (Urdu), we used the the *urduhack/RoBERTa-urdu-small* model from the Hugging Face hub, and fine-tuned two versions of it it using the method described above.

Finally, to further improve the classification step, we implemented ensemble voting. Our ensemble voting strategy involved taking the best three models from our existing system, and implementing a voting system to get optimized predicted labels and thereby increase our Macro-F1 score. We ran these best three models, got their predictions, and then used a voting strategy to get the best predictions. We used a majority-vote system; so if two out of the three models voted for sadness, then that was the prediction to use. However, if all three models predicted different labels, then we reverted to the first model in the list as the tiebreaker.

5 Results

5.1 Primary Task

	BoW	EmoBoW	W2VPT
initial sys			
SVM	13.7	15.9	7.6
DT	24.4	21.6	20.3
improvements			
DTb	25.5	22.7	14.4
N+DT	28.1	24.1	14.9
N+DTb	21.9	25.2	13.7
B+DT	26.8	**29.9**	14.6
B+DTb	24.6	22.2	13.9
N+B+DT	26.4	25.9	13.1
N+B+DTb	22.3	24.6	13.2
	RoB.		
RoBERTa	**35.7**		
B+RoB.	**45.9**		

Table 8: **Ablation Study:** Primary task macro F1 scores for initial systems and various improvement component configurations on the development set: *N*: Negation, *B*: Balancing, *DTb*: Decision Tree Boosting. **Bold** text denotes the three highest performing configurations. An underline denotes the highest performing configuration overall. Bag of words embeddings are used in baseline systems.

As seen in Table 8, our best primary task initial system was a decision tree classifier paired with the baseline bag of words vectors. It yielded a weak macro F1 score of 24.4. Un-boosted decision tree and SVM classifiers that did not handle class imbalance or negation yielded low performance even on their best vector configurations. In particular, the majority of SVM system predictions overly favored the majority class, leading to misleadingly high values for accuracy.

In addition to initial results, Table 8 also displays an ablation study showing the individual impact of various improvement components (Negation, Balancing, Decision Tree Boosting, and RoBERTa) on our system. Overall, fine-tuning a RoBERTa model on our own balanced training data led to the largest improvement in our initial primary system, yielding a macro F1 of 45.88. When data is not balanced, we produce our second best macro F1 (35.7). Our third best macro F1 (29.9) is produced by a traditional machine learning system that uses emotion enhanced bag of words vectors and a decision tree classifier alongside balancing the training dataset.

The balancing strategy we selected was random over/under sampling to the average class size. Our rationale for this decision can be gleaned from the results in Table 9 which displays the macro F1 scores that resulted from trials of each class balancing strategy on our four vector types and the decision tree classifier[8]. Random over/under sampling performed best in 3 out of 4 cases–highest with emotion enhanced bag of words vectors using decision trees–and was selected to help improve our system.

Vector	Over/Under	SMOTE	S.Tomek
BoW	**26.81**	25.24	23.9
Emo	14.64	**16.36**	15.9
EmoBoW	**29.9**	23.06	24.67
W2VPT	**14.56**	14.2	14.45

Table 9: Primary task macro F1 scores for class balancing strategies on the development set using *decision tree* classifier. **Bold** text denotes the highest performing balancing strategy for each vector type. An underline denotes the highest performing configuration overall.

As a final improvement to our system we conducted ensemble majority voting on our three previously described best systems. Using the unbalanced RoBERTa model as a tie-breaker produced a higher macro F1 on the development data (50.26) as compared to using the balanced version as a tie

[8] Almost all trials on the SVM classifier performed worse than using the decision tree classifier.

breaker (47.17). The former ultimately produces the best system for our primary task. As seen in Table 10, running our best system on the development data leads to a macro F1 score of 50.26. 68.62% of predicted emotions are true positives (macro precision) and 47.96% of emotions that *should* be predicted are detected (macro recall). The system is 62.96% accurate (micro precision/recall/F1). Although, when performing inference on our test dataset, each of these metrics decrease noticeably, possibly indicating overfitting.

Data	Mac F1	Mac R	Mac P	Acc
Dev	50.26	47.96	68.62	62.96
Test	36.97	37.91	39.83	54.46

Table 10: Primary task macro F1-score, macro recall, macro precision, and accuracy using ensemble majority voting on the best three systems, where accuracy is equivalent to micro F1, recall, and precision. Results on development and test data are shown.

	A	D	F	J	N	Sa	Su
A	**21**	6	0	0	3	6	0
D	5	**7**	0	0	0	6	0
F	4	1	5	0	2	**9**	0
J	0	0	0	0	4	**6**	0
N	6	0	2	0	**17**	6	0
Sa	3	0	5	1	2	**64**	2
Su	4	0	4	0	**5**	**5**	1

Table 11: Confusion matrix for primary task test predictions. 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 in **bold**.

Performance for each emotion class is displayed as a confusion matrix in Table 11. We see that 4 of the 7 emotion classes most often return true positive predictions–three of these are the most frequent classes: sadness, anger, and neutral. The other class is disgust–the penultimate least frequent. Otherwise, emotions are likely to be falsely predicted as sadness, the majority class. These such lowest performing emotion classes are fear, joy and surprise. No joy essays are properly classified.

5.2 Adaptation Task

As seen in Table 12, the gain from fine-tuning a pretrained model as compared to using traditional machine learning methods and a baseline in our adaptation task is minimal. Additionally, the difference in performance between fine-tuning on balanced versus unbalanced training data is miniscule. By a small margin, the best system for our adaptation task is the unbalanced RoBERTa model with a macro F1 score of 52.92. This is suprisingly better than our best primary task system.

System	Mac F1
BoW DT	47.6
EmoBow DT	50.54
RoBERTa	**52.92**
Balanced + RoBERTa	52.68

Table 12: Adaptation task macro F1-score on bag of words baseline, traditional machine learning, and fine-tuned pretrained model approaches.

As seen in Table 13, when running our best system on the development data 76.01% of predicted emotions are true positives (macro precision) and 51.3% of emotions that *should* be predicted are detected (macro recall). The system is 84.96% accurate (micro precision/recall/F1). These metrics decrease slightly when the system is ran on the test dataset.

Data	Mac F1	Mac R	Mac P	Acc
Dev	52.92	51.3	76.01	84.96
Test	48.43	48.51	67.39	83.81

Table 13: Adaptation task macro F1-score, macro recall, macro precision, and accuracy on the development and test data using the unbalanced fine-tuned RoBERTa model. Accuracy is equivalent to micro F1, recall, and precision.

Performance for each emotion class is displayed as a confusion matrix in Table 14. As in the primary task, we see that 4 of the 7 emotion classes most often return true positive predictions–all four are the most frequent classes: neutral, sadness, joy, and surprise. No disgust tweets are properly classified.

6 Discussion

Our best-performing system for our primary task is the ensemble voting strategy used on our three best non-ensembled systems (balanced RoBERTA, unbalanced RoBERTa, and balanced EmoBoW vectors with DT), yielding a macro F1 on the development data of 50.26. This is an improvement on our best non-ensembled system's macro F1 of 45.88,

	A	D	F	J	N	Sa	Su
A	1	0	0	5	0	1	**12**
D	2	0	0	0	0	0	**3**
F	0	0	3	**11**	0	5	1
J	0	0	1	**64**	0	20	11
N	0	0	0	0	**371**	0	0
Sa	0	0	2	13	0	**101**	5
Su	1	0	0	6	1	14	**49**

Table 14: Confusion matrix for adaptation task test predictions. 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 in **bold**.

and points to the usefulness of ensembling even when using supposedly sophisticated pretrained language models.

The approach of fine-tuning a RoBERTa model also performed surprisingly well for this initial attempt on our adaptation task, earning a macro F1 of 52.92 on the development data - even better than performance on our primary task. This may be due to the larger size of the training data for our adaptation task (5463 examples for adaptation task, compared to 1647 for primary) and the larger size of the pretrained model we used for our adaptation task (125M parameters for adaptation task model, compared to 82M parameters for primary). However, it is also of note that the RoBERTa model only performed marginally better (macro F1 of 52.92) than the best Decision Tree model (macro F1 of 50.54) on the adaptation data.

Performance on the test data was lower, particularly for our primary task (macro F1 of 50.26 on dev, compared to 36.97 on test). This may indicate overfitting, particularly since we have developed several iterations of our primary system on the development data. This effect was not as pronounced for the adaptation task (macro F1 of 52.92 on dev, compared to 48.43 on test).

6.1 Qualitative Analysis of Primary Task

A qualitative analysis of the predictions for English essays reveals some patterns to help us understand the performance of our system. Our final system for D4 displays similar error patterns compared to our prior D3 system.

6.1.1 Correct predictions

Our system consistently makes the correct predictions for majority class labels in which the emotion is stated clearly in the text of the essay, as in the following examples:

- *That's incredibly disgusting* (disgust)
- *I am really saddened by what is happening in Brazil.* (sadness)
- *It is scary that we can not peacefully protest in this country without fear of injury or arrest* (fear)

6.1.2 Incorrect predictions

Incorrect predictions made by our system tend to fall in four categories: multiple emotions described in the text of the essay, nuanced expression of emotion, minority class emotions, and poor annotation of the data.

1. Many of the articles appear to have inspired multiple emotions for the participants. For example, some essays contained expressions of both fear and sadness, fear and surprise, or sadness and joy. Our system would often identify an emotion that, while expressed in the text, was not the primary emotion of the essay. In the following example, the gold annotation is *disgust*, but the author also expresses sadness, which likely led to our system predicting *sadness*.

   > *I found this article and thought you might be interested in it. This is so sad and disgusting. Gun violence really must be stopped in this country and it's hurting innocent people and even unborn people around this country. I think there might be a die in we can participate in happening this week if you can come*

2. As stated earlier, our system did very well classifying essays which stated clearly ”I felt ...” or ”This article made me feel..”. However, essays which did not contain specific definitions of emotion were more challenging for the system to classify. The following essay, which is labeled with the emotion ”disgust”, does not provide such a clear definition of the author's emotion and seems to contain some sarcastic passages. The gold label for the following essay is *disgust* and our system classified it as *anger*.

> *Funny how it always comes down the to the gun and not the person wielding it. People will always be able to get guns, legally or not. the issue is more with society treatment of young men. Too many single moms, too many kids being ostracized and bullied. But yeah, that's the gun's fault. i don't own a gun, but I'd be damned if I gave one up to those liberal fascists.*

3. Our system had challenges identifying the least frequent emotion, *joy*, even when this emotion was clearly expressed in the text as in the following essay. Our system incorrecntly classified this as *sadness*.

> *I am really touched by this story. the rescue just restored my faith in humanity.The success of the rescue operation proves that extraordinary things can be achieved with passion and determination. human conflict causes suffering to both human beings and animals. Suffering animals should not be forgotten or overlooked, even in the midst of human conflicts or natural or man-made disasters. I was very touched and encouraged by these acts of bravery and care towards animals.*

4. Finally, there were essays which (in our opinion) were mislabeled by the annotators. While our system assigned what we believe to be the correct label, it did not match with the gold label and so contributed to our error rate. The following essay has the gold label of *fear*, but our team feels it is more accurately represented by the label of *disgust*.

> *This article about a firecracker going off during a protest is about the stupidest thing I have seen in a while. The journalists are pond scum trying desperately to write anything they can to divide people up. College kids can't bear a firecracker going off, they need to institute mandatory military service like Israel does, then maybe they would not be scared chickens crying about everything.*

6.2 Qualitative Analysis of Adaptation Task

Our adaptation task focused on Urdu language data, specifically, Urdu language tweets. Our performance on the adaptation task was better than on our primary task, so there were a number of correct predictions, but there were still several incorrect predictions. The common thread between the correct and incorrect predictions was having an explicitly named emotion in the sentence. This could help the model, if that named emotion was the same emotion of the overall sentence, but it could also hurt the prediction if that named emotion was different from the overall sentence's emotion.

6.2.1 Correct predictions

Our system performed well on our adaptation task, making a number of correct predictions. Many of the correct predictions were due to the fact that the gold standard emotion label was often named in the sentence. For example:

- (Translated): *Your body will adapt to it for a few days, but the sadness will still be there, but it is new.* (sadness)
- (Translated): *The joy of Ramadan* (joy)

In both of the above examples, the emotion of the overall sentence is explicitly named in the sentence, resulting in a correct prediction.

6.2.2 Incorrect predictions

While our system performed better on our adaptation task than it did on our primary task, there are still some incorrect predictions. One type of incorrect prediction occurs when there is an emotion word named in the sentence that is different than the overall emotion of the sentence. For example, the following sentence:

- (Translated): *This sadness and despair does not affect you* (surprise)

As one can see in this example, because sadness is mentioned by name in the sentence, the predicted emotion from our model was sadness. However, the gold standard labels it as surprise, as the overall emotion of the sentence is more surprised than it is sad.

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 experienced 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.

A further risk introduced by working on Urdu in the adaptation task is the risk of the authors (some of whom do not speak Urdu) misinterpreting the results, and this risk would be further amplified if the system were to be deployed in a real use context.

7.2 Limitations

Our system performance improved throughout the course of the project, and we experimented with a wide variety of techniques. However, a major limitation of our system was the separate pipelines that we developed for traditional machine learning classifiers and pre-trained language models. We first developed a sophisticated pipeline for our traditional classifiers that included several linguistically-informed preprocessing and embedding techniques. Only later did we introduce a pretrained language model, and were obligated to develop a separate pipeline for it that did not take advantage of our previous strategies. Blending these two approaches may have produced an even better performing system.

8 Conclusion

In the development of a multi-class emotion classification system on English essays, we experimented with traditional machine learning approaches, fine-tuning pre-trained language models, and employing ensemble voting on our best systems. We found that balancing our unbalanced dataset helped boost our performance for both traditional machine learning models and a fine-tuned RoBERTa language model. Leveraging the RoBERTa model provided us with the largest gains in performance. Ensembling our three best systems also proved to be a successful strategy for further gains. Nonetheless, our emotion classes with the least training instances failed to show satisfactory predictions while our majority classes showed more promising results.

We were able to successfully adapt our system to perform multi-class emotion classification on tweets from a low resource language, Urdu. We leveraged both traditional machine learning approaches and again fine-tuned a pre-trained RoBERTa model given its strength in our primary task. Both methods and a baseline showed similarly decent performance, with the pre-trained model reigning supreme. Ultimately, the performance of our adaptation system met and slightly exceeded the performance of our primary task. This result and the difficulty to see gains beyond a baseline is potentially due to the larger dataset available for our adaptation task. Still, we faced a recurrent issue of only the most frequent emotion classes showing high prediction performance.

Overall, both of our systems saw improvement with the inclusion of more advanced NLP techniques but fell prey to the quality and prevalence of data. One solution that could enrich our data resources is data augmentation. In future emotion classification tasks, researchers should ensure that data is properly annotated and rich with the content to train on each emotion class.

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.

Nastaran Babanejad, Ameeta Agrawal, Aijun An, and Manos Papagelis. 2020. A comprehensive analysis of preprocessing for word representation learning in affective tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5799–5810, Online. Association for Computational Linguistics.

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.

Gustavo Batista, Ana Bazzan, and Maria-Carolina Monard. 2003. Balancing training data for automated annotation of keywords: a case study. pages 10–18.

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.

Nitesh Chawla, Kevin Bowyer, Lawrence Hall, and W. Kegelmeyer. 2002. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res. (JAIR)*, 16:321–357.

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.

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.