NLP - HW 1 (Written Portion

Uttara Ravi

1 Performance

Model 1 : Ngram

Target	F1-score
0	0.31
1	0.52
2	0.65
macro	0.49

Model 2 : Ngram+Lex

Target	F1-score
0	0.37
1	0.53
2	0.62
macro	0.51

Model 3 : Ngram+Lex+Enc

Target	F1-score
0	0.38
1	0.53
2	0.61
macro	0.51

Model 4 : Custom

Target	F1-score
0	0.38
1	0.53
2	0.62
macro	0.51

2 Error Analysis

(a) I chose the **All caps** (the number of words with all characters in upper case) and the **Hashtags** (the number of hashtags) feature. In online posts, capital letters are used to indicate

emphasis of content, hence capital letters tend to carry a lot of sentiment in them. More polarized tweets will contain more all-capital words. **Hashtags** are certain words in tweets that are specially marked with a hashtag (#) to indicate the topic or sentiment.Hashtagged emotion words such as joy, sadness, angry, and surprised are good indicators that the tweet as a whole (even with- out the hashtagged emotion word) is expressing the same emotion. The count of these hashtagged words hence would help in indicating the degree of polarity. **Elongated words** features would also be a good pick, which is why I skipped it here and chose it as my custom feature. I felt that **POS tagging** wouldn't be very useful because just knowing the POS tags doesn't help us gauge the sentiment. For example, verbs like hate and love capture the opposite sentiment, but this information wouldn't be captured in the POS tagging. They'd both be classified as "V". Another example with a different POS (adjective) is "terrific" and "terrible".

(b) My worst performing model was the Ngram model. But my other three were almost the same. The precision is very marginally higher in my Ngram+Lex model. Lex features are extremely powerful as they carry very accurate measure of the polarity of the word in the document with the help of scores. As expected (and discussed in the paper) the encoding and custom feature didn't improve the performance, because most of the information they carry about the polarity can already be found through ngrams and lex features.

(c) I decided to choose the feature that contains the number of elongated words (and punctuation marks) in a tweet as my custom feature. This is because they help us understand the degree of a certain polarity. The way I implemented the elongated words feature also captures repeated punctuation marks (eg: !!!!) as I am simply checking if a character appears more than 2 times in a row. Repeated punctuation can express extra emotional strength. This in combination with information about previously established positive or negative words in the tweet can help classify it better.

3 Understanding the Features

(a) With the help of character ngrams, we can find elongated words, this is because in the english language we don't find a letter repeated more than 2 times. We can define character grams to check for more than three repitions of a character eg "happyyyy". If we tokenize the tweet without getting rid of punctuation we can use character ngrams to get meaningful patterns out of our emoticons. In the English language no two punctuations appear back to back. But in emojis we have ":)" ";)" etc. Hence character ngrams can be used to find and get probabilities of ngrams. Similarly, we can capture hashtagged words as well by considering a character grams and checking for the occurrence of the # symbol before alphabetical characters.

(b) 1. All caps : Capital words are used to convey a higher degree of polarity. Many All capital words suggest high polarity.
Eg: "I am EXTREMELY ANGRY about the situation" is more polarized than "I am angry"
2. Elongated words and punctuations, such as exclamation and question marks are used to emphasize emotions."I am reallllllyyy dissapointed in tonight's game." conveys a much stronger sentiment than "I'm really disappointed in tonight's game." "what just happened???!!!" also contains a repeated punctuations which is more polarizing than "what just happened?"
3. POS tagging : Highly polarized tweets tend to have more adjectives, whether it is positive or negative. Example: "This is the most obnoxious and terrible movie made in recent times." vs "What is the weather like today?" This information in conjunction with other fea-

tures can help classify the tweet. POS tags on their own however, might not be very helpful. 4. Hashtags : Certain words in tweets are specially marked with a hashtag (#) to indicate the topic or sentiment.In the twitter algorithm, most of the hashtag words used by users tend to capture a lot of sentiment or describe the context of the tweet. Hashtags are usually never insignificant like "#okay" or "#weather". Hence, a more polarized tweet is likely to have more hashtags which is indicated by the number of hashtags.

(c)News articles tend to have all grammatically correct words which appear in the dictionary. However, tweets contain things like Slang (eg: lol, rofl,lmao), elongated words words or punctuations (eg: sooooo, whatttt, !!!!). Hence, it is more useful to have character grams as features in tweets than in news articles.

(d) Non contiguous ngrams might be used in situations where the the word that is replaced by an asterisk might be easily substituted by similar words. Consider the example "a really good movie" and "a very good movie". They both have the same meaning, but they will be classified as two different ngrams. If we use the non-contiguous ngram "a * good" it can capture both of these very similar ngrams mentioned above. Another use is for misspelled words appearing in tweets. "waiting in long queues outside the store" and "waiting in long ques outside the store" if we use the non-contiguous ngram "long * outside" this will help us capture the tweet which will further help the model.

(e) The lexicon in 2.2.1 is derived from common hashtags on twitter, whereas the lexicon in 2.2.2 is derived from common emoticons that appear on twitter. Examples for 2.2.1 - "The new Apple M1 laptops are simply out of this world #trulyAmazed" will correctly be classified as a positive tweet. But sarcasm will not be classified well. Let's say a team played terribly in a game, then a sarcastic or negative tweet will not be classified correctly. "The XYZteam couldn't have played better last night #trulyAmazed" Example for 2.2.2 - "I loved last night's concert :D" will correctly be classified as a positive tweet due to the emoji. But like with hashtags, people can also express sarcasm through emoticons. For example, "A new strain of covid? 2020 is a gift that doesn't stop giving :)" Here due to the "smiling face emoticon" it will be classified as positive, even though it's not.

4 Smoothing

We need to apply some sort of discounting to the counts/probabilites of the ngrams. We can apply a basic formula where we modify all the counts like so:-

$$c_i\prime = c_i \times \frac{N}{N+V}$$

OR subtract it by a constant number. After this we can also reduce all the Maximum Likelihood Estimators to get a valid Probability function that will sum up to 1.

$$P1 = P_{MLE}(w_i|w_{i-1}, w_{i-2}) = \frac{c(w_{i-2}, w_{i-1}, w_i) - d}{c(w_{i-2}, w_{i-1})}$$

Similarily use this "d" to modify the PML function for P2 and P3. But in the case of P3 add the value "d" because that is the case when counts for the bigram and trigram are zero