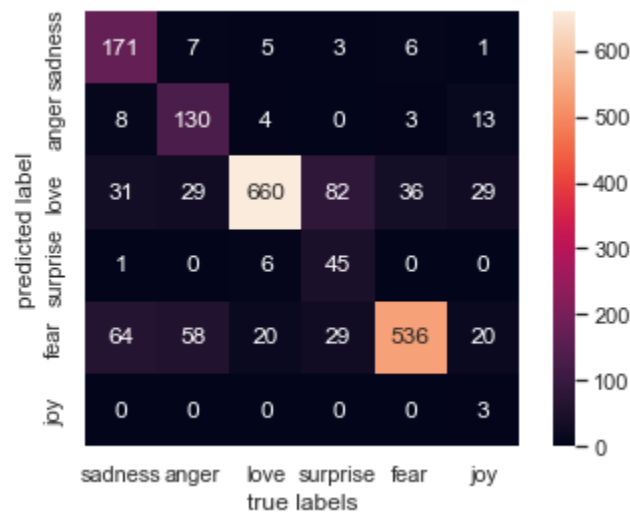


Isabel:

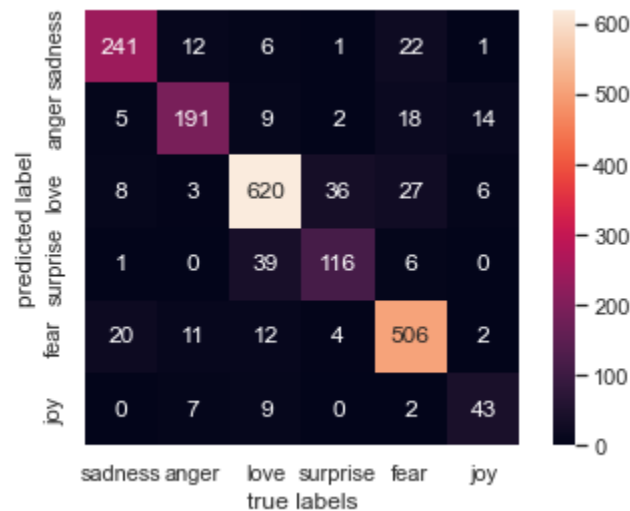
Best models: (found best parameters by GridSearchCV() )

- Naive Bayes: 77% accurate when using CountVectorizer and removing stopwords.  
Parameter for the model: alpha = 1
  - Can see from matrix it misclassified a lot as love or fear



	precision	recall	f1-score	support
anger	0.89	0.62	0.73	275
fear	0.82	0.58	0.68	224
joy	0.76	0.95	0.85	695
love	0.87	0.28	0.43	159
sadness	0.74	0.92	0.82	581
surprise	1.00	0.05	0.09	66
accuracy			0.77	2000
macro avg	0.85	0.57	0.60	2000
weighted avg	0.79	0.77	0.75	2000

- SVM w/ linear kernel: 86% accurate when using TFIDF and CV and removing stopwords  
(both these had the same accuracy and confusion matrix)



	precision	recall	f1-score	support
anger	0.86	0.86	0.86	275
fear	0.81	0.84	0.83	224
joy	0.87	0.92	0.90	695
love	0.79	0.67	0.73	159
sadness	0.91	0.89	0.90	581
surprise	0.73	0.61	0.66	66
accuracy			0.86	2000
macro avg	0.83	0.80	0.81	2000
weighted avg	0.86	0.86	0.86	2000

- Logistic Regression: 88% accurate when using TFIDF and removing stopwords.  
Parameters for the model: C=10 and solver = 'liblinear'



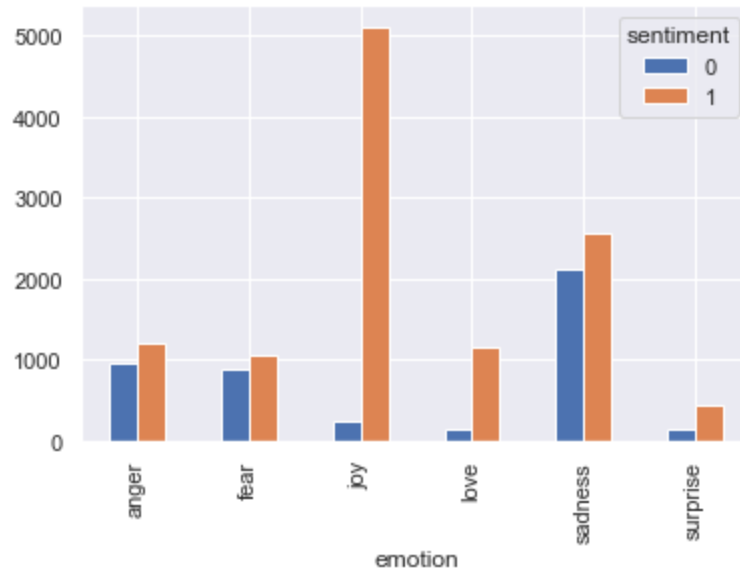
	precision	recall	f1-score	support
anger	0.86	0.85	0.86	275
fear	0.86	0.81	0.83	224
joy	0.89	0.93	0.91	695
love	0.77	0.71	0.74	159
sadness	0.90	0.92	0.91	581
surprise	0.80	0.65	0.72	66
accuracy			0.88	2000
macro avg	0.85	0.81	0.83	2000
weighted avg	0.87	0.88	0.87	2000

Overall, for the models the difference between keeping/including stopwords and using different vectorizers didn't make a significant difference, but the 'best' models were those that removed stopwords. Logistic regression had better metrics across the board.

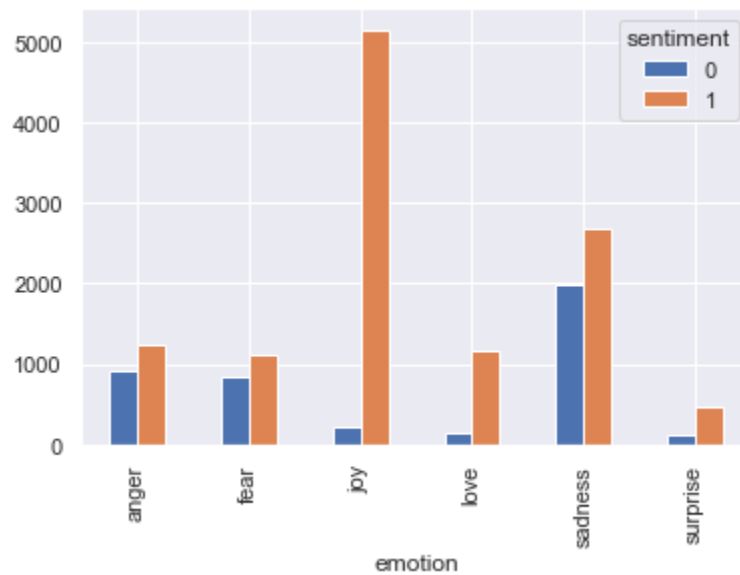
Sentiment Analysis:

- Method: NLTK: 1 = positive, 0 = negative
  - With stopwords:

- Love, joy, and surprise are mostly classified as positive. Other emotions were more of a mix



- Without stopwords: Similar for positive emotions. This time negative emotions were classified more as positive than before



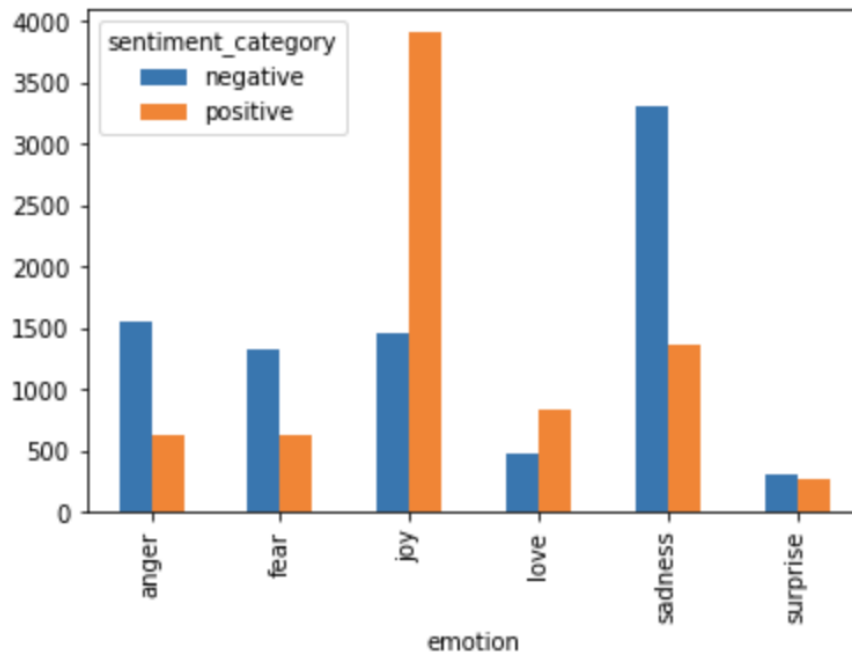
- Conclusion: stopwords add value to sentiment analysis with NLTK

From Taya:

- method: textBlob: positive:1, negative:0

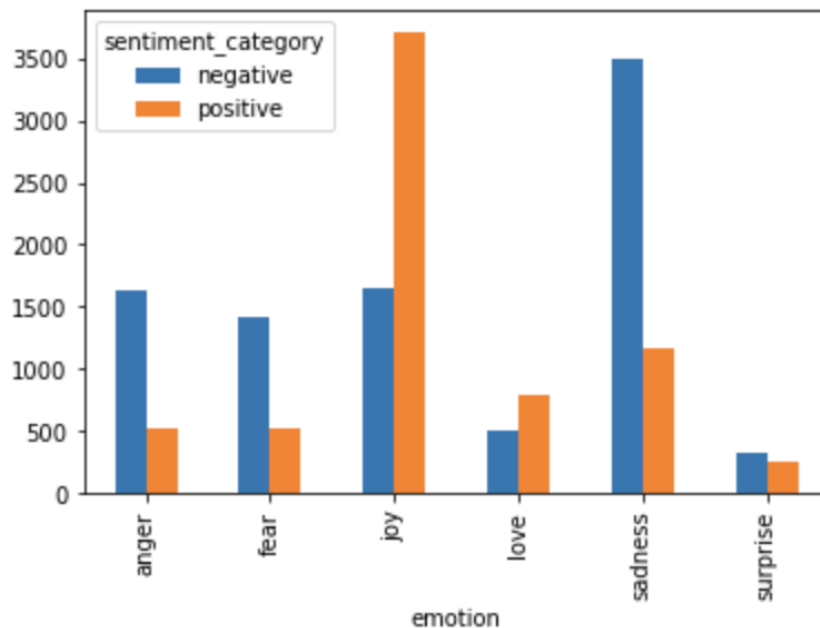
with stopwords:

- joy and love are classified as positive
- surprise is neutral, with roughly equal proportions for positive and negative
- anger, fear and sadness are negative



without stopwords:

- roughly the same proportions as with stopwords, higher results for negative counts.
- 



conclusion: removing stopwords didn't have significant effect

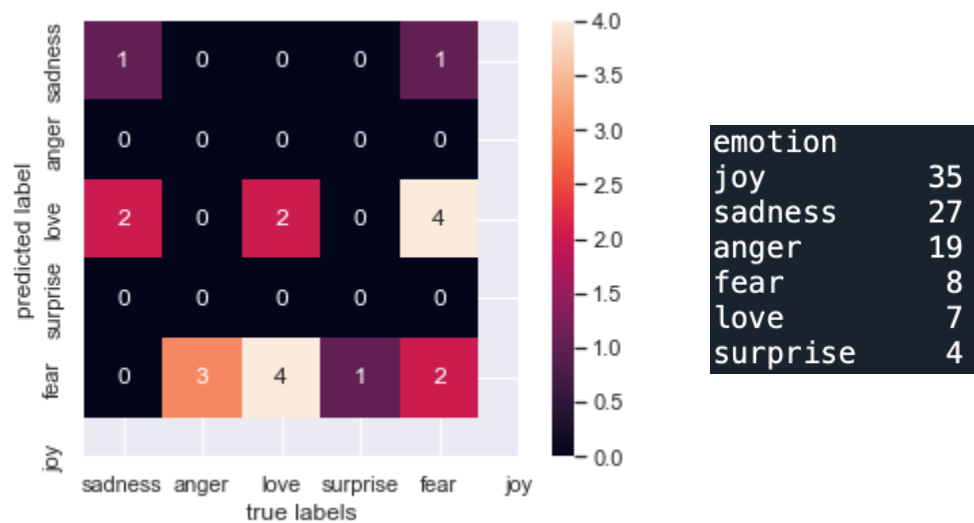
Comparing NLTK and textBlob:

- textBlob classified more instances as negative compared to NLTK

Back Translation: translated from English to Korean then back to English and then cleaned text and tested

- Since I found Logistic regression was the best model, that was the model I used for back translation
- Back translation took a lot of time on computer and couldn't take on as many rows that were in the dataset
  - The first 100 rows from the training dataset were used for training
  - The first 25 rows from the testing dataset were used for testing
  - Accuracy for back translated model was 35%

- Accuracy for original dataset (still only using 100 training rows/20 testing rows) was 25% (right image shows training label counts)



- Accuracies bad probably because lack of data and very unbalanced training labels
- Hard to tell the impact of back translation since we couldn't run it on the entire dataset