Project 2: Text Classification -Manav Bilakhia

Introduction:

This report evaluates the performance of various text classification algorithms including Naive Bayes, logistic regression and other alternatives, using different feature sets. The performance of the classifiers is measured using accuracy, precision, recall, and f-score on the training and development datasets. The best classifier and feature set are selected and used to perform error analysis on the development data. The classifier is then trained on the combined training and development datasets and tested on the unlabeled testing data.

Table 1: Evaluation matrix for the best classifier	
training data	
Accuracy	75%
Precision	70%
Recall	72%
F-score	71%
development data	
Accuracy	78%
Precision	75%
Recall	76%
F-score	75%

Evaluating features:

- All Complex Baseline: The first baseline categorizes all words as complex. The evaluation matrix is as follows:

Table 2: All complex	
training data	
Accuracy	43%
Precision	43%
Recall	100%
F-score	60%
development data	
Accuracy	44%
Precision	44%
Recall	100%
F-score	61%

- Word Length Baseline: This model predicts word complexity based on word length. The best threshold was found to be 7 characters. The evaluation matrix is as follows:

```
Best threshold: 7
training data:
Accuracy: 57%
Precision: 67%
Recall: 0%
F-score: 0%
development data:
Accuracy: 70%
Precision: 62%
Recall: 86%
F-score: 72%
```

- **Word Frequency threshold: This works similar to the length threshold but this time there is a threshold on frequency instead. we tried all thresholds that** satisfy the below inequality:

$$min_{count} < (max_{count} - min_{count}//10000) < max_{count}$$

Word Frequency Threshold: This model predicts word complexity based on word frequency. The best threshold was found to be 28426132. The evaluation matrix is as follows:

```
max ngram counts: 47376829651
min ngram counts: 40
Best threshold: 28426132
training data:
Accuracy: 62%
Precision: 53%
Recall: 87%
F-score: 66%
development data:
Accuracy: 66%
Precision: 57%
Recall: 89%
F-score: 70%
```

Classifier:

- Naive Bayes Classifier: The Naive Bayes Classifier was used with the word length and frequency features and tested on the development data.
- Logistic Regression Classifier: The Logistic Regression Classifier was used with the word length and frequency features and tested on the development data.
- Other classifiers such as SVC, Decision Tree Classifier and Random Forest Classifier were also tested, along with two new features, syllable count and wordnet synonym count. The results of various permutations and combinations of these five classifiers, using different features, are shown in Table 3.

Table 3: Results				
	with syllable and wordnet	**with syllable and without wordnet**	without syllable and with wordnet	without syllable and without wordnet
clf:	GaussianNB()	GaussianNB()	GaussianNB()	GaussianNB()
training data				
Accuracy	62%	57%	60%	55%
Precision	53%	50%	52%	49%
Recall	95%	98%	96%	98%
F-score	68%	66%	67%	65%
development data				
Accuracy	64%	57%	62%	55%
Precision	55%	51%	54%	50%

Recall	98%	99%	98%	98%
F-score	71%	99%	98%	98%
clf:	LogisticRegression()	LogisticRegression()	LogisticRegression()	LogisticRegression()
training data				
Accuracy	74%	74%	75%	74%
Precision	72%	73%	72%	72%
Recall	65%	61%	66%	65%
F-score	68%	66%	69%	68%
development data				
Accuracy	78%	77%	78%	77%
Precision	78%	79%	78%	76%
Recall	69%	66%	70%	70%
F-score	73%	72%	73%	73%
clf:	SVC()	SVC()	SVC()	SVC()
training data				
Accuracy	74%	73%	75%	74%
Precision	70%	70%	70%	69%
Recall	71%	65%	72%	69%
F-score	70%	67%	71%	69%
development data				
Accuracy	77%	76%	78%	77%
Precision	74%	75%	75%	74%
Recall	74%	68%	76%	74%
F-score	74%	72%	75%	74%

clf:	DecisionTreeClassifier()	DecisionTreeClassifier()	DecisionTreeClassifier()	DecisionTreeClassifier()
training data				
Accuracy	99%	99%	99%	99%
Precision	99%	99%	99%	99%
Recall	98%	98%	98%	98%
F-score	99%	99%	98%	98%
development data				
Accuracy	71%	72%	72%	72%
Precision	67%	69%	68%	70%
Recall	65%	64%	70%	64%
F-score	66%	66%	69%	67%
clf:	RandomForestClassifier()	RandomForestClassifier()	RandomForestClassifier()	RandomForestClassifier()
training data				
Accuracy	99%	99%	99%	99%
Precision	99%	99%	99%	99%
Recall	99%	98%	98%	98%
F-score	99%	99%	98%	98%
development data				
Accuracy	76%	73%	75%	75%
Precision	73%	70%	73%	73%
Recall	70%	66%	68%	68%
F-score	72%	68%	70%	70%

Of all the combinations we tested, the best classifier was the SVC using features: word length, word frequency, without syllable count and with a count of synonyms from WordNet. The evaluation matrix for this combination is displayed in the table below

clf	SVC()
training data	
Accuracy	75%
Precision	70%
Recall	72%
F-score	71%
development data	
Accuracy	78%
Precision	75%
Recall	76%
F-score	75%

After conducting the evaluation, we discovered that 176 words were mislabeled. We generated a list of all the incorrectly labeled words, which is as follows:

pawned	assess	nuclei	dwells
clash	stiffen	chicagos	feast
searched	hassles	resume	capture
defying	hyperion	processes	magazine
disconsolate	department	blessing	first-round
everything	shopping	behavior	belting
attended	includes	triage	handful
consumption	lobbed	artists	downed
democracy	elephant	recruit	injuries
grooms	rebuff	generated	fulfill
rash	sweeping	genres	companion
unlikely	contained	lifetime	computer-animated
emptier	facade	cyber	kickoff
stripped	dinners	beige	funding
spontaneous	fraud	remaining	imposes
embody	increases	flurry	home-cooked
elections	monster	husbands	according

assist	vivid	grandsons	photographs
northern	understanding	eighth-grader	somber
scientists	lone	operational	motto
citizen	considers	strut	scientific
veterans	demand	procedural	tatty
hillsides	argument	politics	websites
five-time	wildlife	carvings	hurling
gathered	banned	expands	brisk
bathroom	ensure	declared	yips
arrives	asylum-seekers	seaweed	20-stamp
beef	nod	extracts	promote
monotonously	activities	barred	low-water
ousted	plucky	tragedy	cheating
offstage	fast-food	respite	stranger
implications	childhood	youtube	jolted
featured	onus	trickier	fighting
languages	census	inequity	operations
long-range	mountain	gentler	festival
protest	gait	supporting	notable
proceedings	unevenly	battled	cracking
governments	digging	issued	minister
fairness	quest	replaced	volleys
attire	curious	deciding	intense
glancing	boulders	taboo	seventh-grader
pylons	anymore	passengers	divisions
combine	head-to-toe	rigid	argued
appearances	cortex	advantageous	triangle

After analyzing the evaluation results, we found that 176 words were mislabeled. We also compiled a list of these incorrectly labeled words. The list of mislabeled words is shown below.

Of all the words in the development dataset, only the ones listed above were not labeled correctly, leading to a mismatch rate of approximately 22%. The majority of these mislabeled words were nouns. However, this may just be due to the high proportion of nouns in the overall list of words rather than any specific issue with the model's ability to categorize nouns. Having evaluated the model, the next step is to train it on both the training and development datasets, then test it on the unlabeled testing dataset.

We combined the training and development files to form a larger training dataset and trained our best classifier on it. Afterwards, we applied this classifier to the unlabeled testing dataset, and the predicted labels are stored in the file "test_labels.txt."