CSCE 5543-STATISTICAL NLP HOMEWORK 2

Mamata Shrestha
010985316

1 Algorithms

1.1 Naive Bayes Classifier

Naive bayes is one of the most popular classification algorithm that is mainly used for multi-class classification problems. It reduces the complexity by making an assumption that features are independent of each other. It gives better performance without much training. In word sense disambiguation problem, we use naive bayes classifier to classify whether given word sense belong to one of many possible senses that word can have depending on the context. Naive bayes treats context words as a bag of words without taking into account of the order and position of those context words. It intergates the information from all context words found in the dataset within a given window. The sense prediction is given by:

$$c_{NB} = argmax_{c_j \in C} P(c_j) \Pi P(X_i/c_j)$$

where,

$$P(c/d) = \frac{P(d|c)P(c)}{P(d)}, P(w/c) = \frac{count(w,c)+1}{count(c)+|V|}, P(c) = \frac{N_c}{N}$$

V is the vocabulary size of context words, count(w,c) is the number of times w occurred in the set for c, and count(c) is the number of training set for c.

2 Data Structure

Hash table is used for counting the occurrence of each token in the train set. Hash table is an abstract data structure that maps keys to a value. It uses a hash function to get an index of the hash table where the value corresponding to a key is stored. Collision occur if more than key has same hash value. Therefore, separate chaining method is used to handle the collisions. The chaining method arranges the values that are mapped to the same index in a list. Besides hash table, dictionary and list is also used to store the counts and words.

3 Text Pre-processing

1. **Convert short form words:**
   Conversion of words like ́ve to have, i'm to I am and n't to not and so on for better tokenization in next steps.
2. **Remove stop words:**
   Since the stops words are common and occur almost equally in every context hence does not provide useful information for the disambuigation, it is useful to remove them. It also helps to reduce the complexity in the program.
3. **Tokenization:**
   Used nltk built-in function to tokenize the input.
4. **Text normalization:**
   All tokens are converted to its lower case so that the counting of token for classification does not become case sensitive.

4 Complexity Analysis

Hash table is used to count the occurrence of each token in each senses training file. The time and space complexity of the hash table is O(n), where n is the number of entries. The programs loops for through all documents in the amazon-reviews.txt file and also through the list of words to extract the context words. The complexity for extracting context word for each window is given by O(n*m) where n is the total number of documents in text file and m is the total number of words for which we are extracting the context words.

It takes 6.7 seconds to create a dataset for each window value. Similary, it takes around 1.2 seconds to train and test for single window data. But in turing machine, it takes much longer, around 20 seconds, to generate the dataset.

5 Running code

The input directory with the *amazon_reviews.txt* file must be in the same directory with the code files.I have hardcoded the input file path in bash script. Run code by specifying the shell script, which runs two code files as follows: *./run.sh*

By executing bash script, first dataset with context words for all pseudo words is created by running *generate_context_word.py* scripts. After this, *naive_bayes.py* scripts perform train and test whose results are saved in the outputs directory. The user will now be asked with the prompt whether they want to run train and test for dataset having equal examples in trainset for both senses. If answered yes, then it will generated result and save it on the outputs directory.

6 Results

The accuracy of model for context windows of size 10 is as follows:

1. For psuedoword nightseat
   Accuracy for class night = 0.17
   Accuracy for class seat = 1.0
   Total accuracy of classifier = 0.493

2. For pseudoword kitchencough
   Accuracy for class kitchen = 0.0
   Accuracy for class cough = 1.0
   Total accuracy of classifier = 0.2

3. For pseudoword carbike
   Accuracy for class car = 0.003
   Accuracy for class bike = 1.0
   Total accuracy of classifier = 0.044

4. For psudoword manufacturerbike
   Accuracy for class manufacturer = 0.73
   Accuracy for class bike = 0.92
   Total accuracy of classifier = 0.837

5. For pseudoword bigsmall
   Accuracy for class big = 0.57
   Accuracy for class small = 0.79
   Total accuracy of classifier = 0.692

6. For pseudoword hugeheavy
   Accuracy for class huge = 0.51
   Accuracy for class heavy = 0.91
   Total accuracy of classifier = 0.724

For the peudowords with the words that are closer in their meaning or context, there is high chance for model to make inaccurate predictions. In our case, both training set and test set size is heavily imbalance in most of the pseudowords. Therefore, the acuracy values are not following the expected pattern that much. Among the pseudowords, manufacturerbike has best and carbike has the least overall accuracy.

The accuracy of model for context windows of size 5 is as follows:

1. For psuedoword nightseat
   Accuracy for class night = 0.22
   Accuracy for class seat = 1.0
   Total accuracy of classifier = 0.52

2. For pseudoword kitchencough
   Accuracy for class kitchen = 0.0
   Accuracy for class cough = 1.0
   Total accuracy of classifier = 0.2

3. For pseudoword carbike
   Accuracy for class car = 0.018
   Accuracy for class bike = 1.0
   Total accuracy of classifier = 0.058

4. For psudoword manufacturerbike
   Accuracy for class manufacturer = 0.78
   Accuracy for class bike = 1.0
   Total accuracy of classifier = 0.89

5. For pseudoword bigsmall
   Accuracy for class big = 0.56
   Accuracy for class small = 0.74
   Total accuracy of classifier = 0.659

6. For pseudoword hugeheavy
   Accuracy for class huge = 0.59
   Accuracy for class heavy = 0.88
   Total accuracy of classifier = 0.741

The accuracy of model for context windows of size 20 is as follows:

1. For psuedoword nightseat
   Accuracy for class night = 0.13
   Accuracy for class seat = 0.98
   Total accuracy of classifier = 0.46

2. For pseudoword kitchencough
   Accuracy for class kitchen = 0.0
   Accuracy for class cough = 1.0
   Total accuracy of classifier = 0.2

3. For pseudoword carbike
   Accuracy for class car = 0.0016
   Accuracy for class bike = 1.0
   Total accuracy of classifier = 0.043

4. For psudoword manufacturerbike
   Accuracy for class manufacturer = 0.608
   Accuracy for class bike = 0.92
   Total accuracy of classifier = 0.77

5. For pseudoword bigsmall
   Accuracy for class big = 0.59
   Accuracy for class small = 0.73
   Total accuracy of classifier = 0.667

6. For pseudoword hugeheavy
   Accuracy for class huge = 0.53
   Accuracy for class heavy = 0.95
   Total accuracy of classifier = 0.747

The accuracy of model for context windows of size 10 and equal no of training set is as follows:

1. For psuedoword nightseat
   Accuracy for class night = 0.517
   Accuracy for class seat = 0.94
   Total accuracy of classifier = 0.68

2. For pseudoword kitchencough
   Accuracy for class kitchen = 0.75
   Accuracy for class cough = 1.0
   Total accuracy of classifier = 0.8

3. For pseudoword carbike
   Accuracy for class car = 0.22
   Accuracy for class bike = 0.92
   Total accuracy of classifier = 0.253

4. For psudoword manufacturerbike
   Accuracy for class manufacturer = 0.52
   Accuracy for class bike = 1.0
   Total accuracy of classifier = 0.776

5. For pseudoword bigsmall
   Accuracy for class big = 0.475
   Accuracy for class small = 0.88
   Total accuracy of classifier = 0.692

6. For pseudoword hugeheavy
   Accuracy for class huge = 0.48
   Accuracy for class heavy = 0.95
   Total accuracy of classifier = 0.72

While training using same number of training example for each word,the accuracy of each word in pseudowords were more balanced. The accuracy improved for the class which had very low accuracy previously. For instance, the accuracy for class kitchen increase from 0.0 to 0.75, and for car class it increase from 0.0033 to 0.22. Both of these word pairs had a very large difference in training set size with their respective word pair. The ratio of word pair (kitchen,cough) was 60:13 and for word pair (car,bike) it was 2424:103. The overall accuracy got better for every model while using equal size training set. Pseudoword carbike has the lowest overall accuracy in this setting as these word pairs are closer in sense.