Assignment 5

Mohammad Karimiabdolmaleki
University of Alberta
Edmonton, AB
karimiab@ualberta.ca

Ali Zamani
University of Alberta
Edmonton, AB
azamani1@ualberta.ca

In this assignment, we have implemented a relation extraction with the help of a classifier that can classify a given sentence to one of the four classes of publisher, performer, director, character.

1 Performance Report

In this section, we will try to evaluate the performance of the classifier by some common metrics like accuracy, mire-average precision, and macro-average precision.

1.1 Accuracy

To evaluate the Naive Bayes classifier, the 3 fold cross-validation is applied on the training data which yielded the average accuracy of **0.891**. Moreover, we trained the Naive Bayes classifier on the whole training data and test it via the test data which the accuracy of **0.885** achieved. Table 1 summarizes the accuracy of the training data and test data.

Table 1: Accuracy on train and test data

Data set	Accuracy
Training data	0.891
Test data	0.885

1.2 Confusion Matrix

To examine the classifier more thoughtfully, the confusion matrix is provided in table 2 which helps to figure out which classes have more classifications and which classes classified well. For instance, the number of ”characters” that misclassified as ”director” is 6.

1.3 Micro & Macro Average Precision

In this subsection, first, the confusion matrix associated with each class is provided. To be more specific, the Confusion matrix for ”characters” is provided in table 3. Moreover, Table 4 shows the confusion matrix of ”director”, while Tables 5 and 6 present the confusion matrix of ”performer” and ”publisher”, respectively. According to the confusion matrices, the **Micro-average precision is equal to 0.885** and the **Macro-average precision is equal to 0.886**.

Table 2: Confusion Matrix

Predicted/Golden	Characters	Director	Performer	Publisher
Charactors	87	6	7	3
Director	6	83	3	0
Performer	4	2	90	3
Publisher	6	3	3	94

Table 3: Class 1 (Characters)

Predicted/Golden	True Characters	True Not
System Characters	87	16
System Not	16	281

Table 4: Class 2 (Director)

Predicted/Golden	True Director	True Not
System Director	83	9
System Not	11	294

2 Justification of Design

The naive Bayes algorithm that we implemented from scratch, uses the input text sentences from the train and test CSV files for training and evaluation. We converted the sentences to Bag-Of-Words representation by using the frequency of the words. In Bag-Of-Words, the only thing that matters is the frequency of the words, consequently, the position and order of the words are ignored. Thus, the probability of the sentences is dependent on the frequency of the words. As mentioned in the perfor-

Table 5: Class 3 (Performer)

Predicted/Golden	True Performer	True Not
System Performer	90	9
System Not	13	288

Table 6: Class 4 (Publisher)

Predicted/Golden	True Publisher	True Not
System Publisher	94	12
System Not	6	288

mance report, the accuracy of the model on the test set is close to the average accuracy of the model on the 3-fold cross-validation on the training data. This suggests that the model is indeed learning and it's not underfitting. As a choice of the model design, we decided to not use other features (e.g., the position information, etc.) since we believe that the model is both performing and generalizing well, and adding unnecessary features may lead to either overfitting or even reducing the performance of the model. Moreover, all of the useful information that the classifier needs to be able to classify the relation of a given sentence is present in the 'token' feature itself. So, our justification is that adding the position features may not be that useful since the head and tail themselves are embedded in the sentence in some way.

We followed the suggestion in the lectures and JM's textbook to ignore the unknown words in the test data and to employ Add-1 (Laplace) smoothing for dealing with the words that are only present in a single class. We also employed some preprocessing techniques to deal with noise and unnecessary tokens in the text sentences. In this way, we did not consider the nonalphanumeric tokens. It not only removes unnecessary punctuation and low-frequency tokens but also lowers the computational cost of the runtime.

3 Error Analysis

Since we used the Bag-Of-Words technique in this assignment, which does not consider the position of words in a sentence, any coexistence of a word in two or more classes can prone the classifier to the problem of misclassifying the sentences. For example, "director" and "characters" classes misclassified 12 times as "characters" or "directors", more than all other classes since words such as "play" are common between two classes. To be more specific,

Table 7: Precision and recall

	Precision	Recall
Charactors	0.845	0.845
Director	0.902	0.883
Performer	0.909	0.874
Publisher	0.887	0.940

consider "He played the fictional Buster Kilrain in Ron Maxwell 's Civil War Duology" which belongs to the "director" class. However, it is classified as "characters" since it has some similar words with the "directors" class.

Based on table 3, the "characters" class is mostly classified incorrectly, since it may have the most common words with other classes. One approach for decreasing common words is using head and tail words for training and testing the model but since it may prone the classifier to overfit the training data, we will not use this approach.

In contrast, according to table 6, the "publisher" class has the least number of mistakes, which returns back to the fact that sentences that are labeled as "publisher" consist of more dates which distinguish them from other classes in training and test data.

In short, adding more training data and features can assist the classifier to achieve higher accuracy. However, adding more features without having enough data may cause overfitting and decrease the accuracy of the test data. In this assignment, since the accuracy is acceptable, the number of training data seems enough.

4 Repository URL

https://github.com/UOFA-INTRO-NLP-F21/f2021-asn5-MMDPY