Lab 2 Naïve Bayes and KNN

SHINE-MING WU SCHOOL OF INTELLIGENT ENGINEERING

Spring 2023

Prerequisites

- You need to have some background knowledge about Naïve Bayes (NB) and K Nearest Neighbors (KNN) classifier. If not, you can check out: NLP_Lec6 and `https://en.wikipedia.org/wiki/Naive_Bayes_classifier` or `https://www.bilibili.com/video/BV1Mh411e7VU?p=10&spm_id_from=333.851.header_right.history_list.click`
- You need to install the NLTK, Pandas, Numpy, Scipy, and scikit-learn packages:

```
pip3 install --upgrade nltk pandas numpy scipy scikit-learn
```

1 Naïve Bayes

Make sure you have the following file(s): `lab2_skeleton.zip`, including:

```
lab2_skeleton
├── lab2_skeleton.py
├── stop_words.txt
├── nlp_lab2.pdf
└── data
    ├── test_NB.csv
    ├── train_NB.csv
    ├── test_KNN.csv
    └── train_KNN.csv
```

Q1 Preprocess the training set refers to the following steps,

1. Use pandas to read data from `data/train_NB.csv` and `data/test_NB.csv`
2. Use nltk to tokenize text into words
3. Turn words into Bag-of-words representation using raw frequency.

Q2 Write code to compute the probabilities.

1. Design the Laplace Smoothing
2. Compute $P(Y = y_i)$
3. Compute $P(x_j|Y = y_i)$

Q3 Write code to predict labels

1. Compute $P(Y = y_i) \prod_j^V P(x_j|Y = y_i)$
   (hint: $P(Y = y_i) \prod_j^V P(x_j|Y = y_i) = \exp(\log(P(Y = y_i)) + \sum_j^V \log(P(x_j|Y = y_i)))$)
2. Compute $P(Y = y_i|x_1, \cdots x_V)$.
3. Predict the label of each documents in the test set, and output the predictions to submission_NB.csv
4. Calculate the precision, recall, and F1 score of each category and average precision recall, and F1 score in the validation set.
5. (Optional) Try other text categorization methods such as Support Vector Machine (SVM), AutoML (AutoGluon `https://auto.gluon.ai/stable/index.html`).

2 KNN

Q1 Preprocess the training set refers to the following steps,

1. Use pandas to read data from `data/train_KNN.csv` and `data/test_KNN.csv`
2. Use nltk to tokenize text into words
3. Turn words into Bag-of-words representation using raw frequency.

Q2 Write code to design a KNN classifier

1. Calculate tf-idf from the data matrix with the following formulas.(K = 0.5)

$$\text{tf}(t, d) = (K + (1 - K)\frac{f_{t,q}}{\max_t f_{t,q}})$$
$$\text{idf}(t) = \log(\frac{N}{1 + n_t}) + 1$$
$$\text{tf-idf}(t, d) = \text{tf}(t, d) \times \text{idf}(t)$$

2. Design the eucledian distances metric between documents.
3. Design the KNN classifier with K = 3 then predict the validation set. Calculate the precision, recall, and F1 score of each category and average precision recall, and F1 score in the validation set.
4. Predict the label of each documents in the test set, and output the predictions to submission_KNN.csv

3 Submission

If you miss onsite assessment, you need to submit four files (program output, `submission_NB.csv`, `submission_KNN.csv` and python script.) to **BlackBoard**. After you finished the assignments, make sure you include the header information in the beginning of your code

```
# author: Your_name
# student_id: Your_student_ID
```

Copy all the program output in to a text file named `StudentID_StudentName_lab2_output.txt`, zip two .csv file named `StudentID_StudentName_lab2_NB.csv`, `StudentID_StudentName_lab2_KNN.csv` and python script solution named `StudentID_StudentName_lab2.py` to `StudentID_StudentName_lab2.zip`. Then sumit the zipped files to **BlackBoard**.