Assignment 3

Due on December 6, 2022 (23:59:59)

Instructions. The goal of this problem set is to make you understand and familiarize with Naive Bayes algorithm.

Classification of News Articles

In this assignment, you will try to determine which category a news article belongs to among five categories (Sport, Business, Politics, Entertainment, Tech) (see Table 1). You will implement a Naive Bayes classifier and verify it's performance on English News Dataset [1]. As you learned in class, Naive Bayes is a simple classification algorithm that makes an assumption about the conditional independence of features, but it works quite well in practice.

ArticleId	Text	Category
651	wales silent on grand slam talk rhys williams says wales are still not thinking ...	sport
2034	car giant hit by mercedes slump a slump in profitability at luxury car maker mercedes has prompted ...	business
1582	howard truanted to play snooker conservative leader michael howard has admitted he used to play ...	politics
1407	rapper snoop dogg sued for rape us rapper snoop dogg has been sued for ...	entertainment
1532	security scares spark browser fix microsoft is working on a new version of its internet explorer web browser. ...	tech

Table 1: Some examples from different categories in the dataset

Dataset
English News Dataset is a dataset provided to determine the category of a news article. It includes the following features:

- **"ArticleId":** Id number of the article
- **"Text":** Text of the article, could be incomplete
- **"Category":** Ground-truth category label of the news article (Sport, Business, Politics, Entertainment, Tech)

You can download your training dataset from **this link**

Approach

1. **Part 1: Understanding the data**
   You will be predicting the category of a news article from words that appear in the headline. Is that feasible? Give 3 examples of specific keywords that may be useful, together with statistics on how often they appear in each of five categories' articles.

2. **Part 2: Implementing Naive Bayes**
   You will represent your data with listed approaches and use them to learn a classifier via Naive Bayes algorithm. You have to implement your own Naive Bayes algorithm.

   - Features: You will use Bag of Words (BoW) model which learns a vocabulary from all of the documents, then models each document by counting the number of times each word appears. You will use BoW with two options:

     – Unigram: The occurrences of words in a document(frequency of the word).

     – Bigram: The occurrences of two adjacent words in a document.

   Note: You should compute the log probabilities to prevent numerical underflow when calculating multiplicative probabilities.
   You may encounter words during classification that you haven't during training. This may be for a particular class or over all. Your code should deal with that. Hint: You can use Laplace smoothing.
   You have to use a dictionary for BoW representation. You can implement your own method to obtain BoW model or you can use Count Vectorizer function (`https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html`).

3. **Part 3:**

   (a) Analyzing effect of the words on prediction

   - List the 10 words whose presence most strongly predicts that the article belongs to specific category for each five categories.

   - List the 10 words whose absence most strongly predicts that the article belongs to specific category for each five categories.

     You can narrow down your dictionary by choosing specific words for articles belongs to all five categories. In other words, your classification results can be improved by selecting a subset of extremely effective words for the dictionary.
     TF-IDF (`https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html`)

and Information Theory are good places to start looking. Reimplement the part2 and see the effect of using specific words on the task.

Compare the influence of presence vs absence of words on predicting whether the article is specific category of five different categories.

(b) Stopwords
You may find common words like “a”, “to”, and others in your list in Part 3(a). These are called stopwords. A list of stopwords is available in sklearn here. You can import this as follows:
`from sklearn.feature_extraction.text import ENGLISH_STOP_WORDS`
Now, list the 10 non-stopwords that most strongly predict that the article belongs to specific category for each five categories.
Analyzing effect of the stopwords: Why might it make sense to remove stop words when interpreting the model? Why might it make sense to keep stop words?

4. **Part 4: Calculation of Accuracy**
You will compute accuracy of your model to measure the success of your classification method:

$$\textbf{Accuracy} = 100 * (\frac{\textbf{number of correctly classified examples}}{\textbf{number of examples}}) \quad (1)$$

Submit

You are required to submit all your code (*all your code should be written in Jupyter notebook* long with a report in ipynb format (should be prepared using Jupyter notebook). The codes you will submit should be well commented. Your report should be self-contained and should contain a brief overview of the problem and the details of your implemented solution. You can include pseudocode or figures to highlight or clarify certain aspects of your solution. Finally, prepare a ZIP file named **name-surname-pset2.zip** containing

- reportcode.ipynb (Jupyter notebook file containing your report and code)
- code/ (directory containing all your codes as Python file .py)

Grading

- Code (50): Part1: 5p, Part2: 25p, Part3: 15p, Part4: 5p
- Report(50): Analysis of the results for prediction: 50p.

  Notes for the report: Preparing good report is important as well as your solutions! You should explain your choices (Unigram, Bigram or both of their use for Bow, or constraints on data) and their effects to the results.

Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

References

[1] https://www.kaggle.com/datasets/qusaybtoush1990/english-news