Medical reports classification

Machine Learning

Hanna Nurska, 499866

June, 2022

I affirm that this report is the result of my own work and that I did not share any part of it with anyone else except the teacher.

1 Data analysis

The dataset includes a total of 3331 medical reports divided into training set of size 2731, validation set of size 300 and test set of size 300. Each of the reports belong to one of 8 categories which are:

- cardiovascular / pulmonary,
- consult,
- gastroenterology,
- general medicine,
- neurology,
- otthopedic,
- radiology,
- surgery.

The number of samples belonging to these classes are: 372 (cardiovascular / pulmonary), 355 (orthopedic), 223 (neurology), 516 (consult), 273 (radiology), 259 (general medicine), 1103 (surgery), 230 (gastroenterology) which is presented in figure 1.

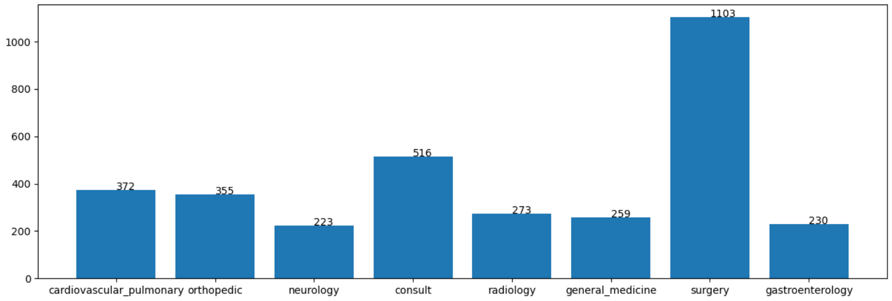

Figure 1: Class distribution

The classes are distributed quite equally except class *surgery* which has considerably more samples. When it comes to train set, number of samples from class *surgery* also exceeds number of samples from other classes. The result is presented in figure 2. This might have influence on classification result, i.e. the class *surgery* might be more probable to choose.

The data is in format .txt, files contain plain text. In order to make use in classification model from this data, it is crucial to extract some features (for example number of occurances of the most popular words) as well as assigned class (we can derive it from the file name). The process and results of data extraction is described in section 2.

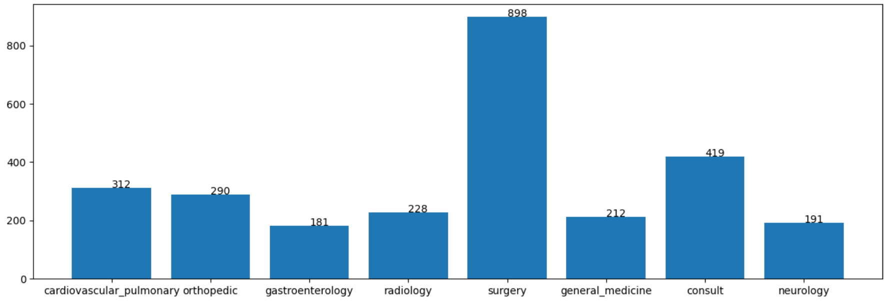

Figure 2: Class distribution

2 Data extraction

First of all, the n most common words occuring in train data were derived and saved into `vocabulary.txt` file. The value n was taken to be 3000. In order to get them, some text preprocessing was applied. To begin with, all punctuation and numbers were removed from the reports. Then, words shorter that 2 letters were deleted, as well as stopwords. Stopwords are words who are really common, which don't have influence on document identification. In other words, stopwords cannot help with recognising the nature of speech. Last but not least, all of the words were stemmed, which means that the endings of the words were cut off. The aim of this operation is to treat the same word in different gramatical forms as one word. The most common words in medical reports from train set were: *patient, right, left, histori, us, place, procedure, normal, pain, year, ...*. As we can see, none of the 10 first words have nautral overtone, i.e. they don't indicate to which category of madicine they might refer. That's why it is important to choose as n big enough value because a big part of common words will occur in many reports, regardless their categories.

After creating a vocabulary, it was used to redesign text reports into *Bag of Words* representation. It's a representation which is a vector of the counters of the occurances of the most popular words (from `vocabulary.txt` file) in the text. The label of the sample was derived from its filename and then, transformed into a number. To do so, all of the classes were saved into a list and then, the index of the particular label was taken. The BoW represantation of all of the reviews and their labels were later saved into a compressed file `train.txt.gz` The data normalization was not necessary since the features are only compared to each other.

At this stage, also validation and test data were converted into BoW representation, normalised and saved into `valid.txt.gz` and `test.txt.gz`, respectively. Of course, the normalization applied to validation and test data was based on *mean* and *standard deviation* values derived from the training set.

For the data extraction part, the code developed in the lab activity was reused.

3 Model selection

Next step is selection of accurate classification model used for the training and after for the evaluation. In order to have wider perspective, two models were implemented, tried and tested. First of the models is Naive Bayes Classifier, described in more details in subsection3.1. As second model, Neural Network was tried out, which is described in subsection 3.2.

3.1 Multinomial Naive Bayes

Naive Bayes is an common model when it comes to text classification. In this approach, as well as in Neural Network approach, each word is treated as feature. The main assumption of Naive Bayes is that all of the features are conditionally independent which means that knowing a class of a sample, the knowledge about one of the features doesn't give any knowledge about any other feature.

In the task of classification of medical rewievs Multinomial model was used because there are more than two classes. The prediction of the Multinomial Naive Bayes is defined by the following formula:

$$\hat{y} = \underset{y \in \{0,\ldots,k-1\}}{\operatorname{argmax}} \left(\sum_{j=0}^{n-1} x_j \log \pi_{y,j} \right) + \log P(y), \tag{1}$$

where x_j is the feature vector, $\pi_{y,j}$ is the learnt probability of the occurance the word j in class y, $P(y)$ is the probability of the class y.

The main objective of the training is to find the probabilities $\pi_{y,j}$.

For the use of the project, the implementation from sklearn was used. It takes as parameters feature matrix X, label vector Y and *alpha* which is smoothing parameter. Different values of parameter *alpha* were tried while training the model.

3.2 Neural Network