**Naïve Bayes classifier**

Is machine learning algorithm using probability theory and Bayes’ rule to predict category of samples (sample could be for example text document and corresponding category would be Art or Sport). For learning it uses bag-of-words representation of text data, which counts the frequency of each word in the training samples for the category they are assigned to. Lets say we have 5 training samples with two different types of categories reviews (categories) of product Good and Bad in the following table:

|  |  |
| --- | --- |
| **Text of sample** | **Category (review)** |
| excellent headset | Good |
| mic doesn’t work | Bad |
| love this headset | Good |
| bad quality mic | Bad |
| great mic | Good |

This data is expressed in the following table by using bag-of-words text representation of samples for each category, where attributes are words and their values are frequencies.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Categories** | **excellent** | **headset** | **mic** | **doesn’t** | **work** | **love** | **this** | **bad** | **quality** | **great** |
| Good | 1 | 2 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| Bad | 0 | 0 | 2 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |

Counting word frequencies is almost the entire learning process, and with this table, we have the largest tool for computing probabilities for each category in which the unknown sample may belong to. To make a prediction of an unknown sample we simply choose category with the highest calculated probability. Samples used for counting word frequencies are known as training dataset.

**Category selection**

In order to see how exactly this Naïve Bayes classifier works, lets head to the web application and in section “Classification parameters and results” there are chosen by default these three categories:

* ohsu-Bacterial\_Infections\_and\_Mycoses
* ohsu-Stomatognathic\_Diseases
* news-Medicine

**Sample selection**

Click on Classify and below will be triangle graph displayed which is representing the probability of some classified unknown samples. We can see that there is one sample belonging to Bacterial\_infections category classified incorrectly to Stomatognathic\_Diseases. If we click on this sample, we will be able to see additional charts through which we can see computational processes of this classification.

The triangle graph on the right side contains all words of the clicked sample and shows their calculated probabilities based on which categories they mostly correspond.

Below these two triangle graphs there is one more graph which is expressing the flow of probability calculations after each word in the unknown sample. Dash lines represent probability of individual words, which can be seen on the right triangle graph. The bold line represents summarizing probability that takes into account all previous words in one segment of words. These segments are separated by bold points and the color of these bold points is telling us into which category certain segment of words would be classified. After these points, the summation probabilities are resetted (it is just for better visualization of summarizing probabilities over the entire sample). The length of the word segments can be changed using the slider above “Reset point of overall probabilities”.

**Words**

Lets set it to 35. At the beginning of second segment we will see how the probability is raising to category that caused incorrect classification (Stomatognathic\_Diseases) of the whole second segment. The most error-causing words in this segment are words: tumors (39th), bony(47th), tumor (49th or 56th) and dissection (66th). These words can also be seen in the last graph below, which speaks of their uniqueness by calculating the percentage difference between the most likely and the second most likely word among categories. Lets take a closer look at word tumor (49th) via attributes of the training dataset which is expressed by bag-of-words model.

**Attributes of training dataset**

Graph of these attributes is shown above (features of training dataset). In order to display the most frequently used words in category that caused classification error, click on the dropdown for selection of this category (Stomatognathic\_Diseases) and lets find word tumor via ctrl+f. We can see that frequency of this word is very small and the ratio difference between categories isn’t that big, but had significant impact on resulting probability of this word. (This word isn’t good visual example of it though, I will state another word shortly which has more visible frequencies and ratio)

We can also see probability results of these words in this graph using the “Word probability” option. Through these two options we can see, how the probability isn’t only influenced by frequencies alone because the value ratios between categories are fundamentally changing.

**Properties of training dataset**

In order to calculate the probability of a word, it is necessary to take into account how many words occur in all samples of each category. Because if we have for example 10 words in one category and 3 of them would be „health“, it is much more likely that word health will correspond more with this category than with category containing 7 such words with 300 total amount of words.

In our case, the difference in the word amounts between categories (Stomatognathic and Bacterial) is approximately 7 times big. This can be seen at the highest graph displayed „Properties of training dataset“. By default, the amount of training samples is set for each category, by using interactive legend we can pick „Words“ to display these amounts of words for each category.

Lets take for example 4. or 5. word (treatment and disease) in the “Features of training dataset” graph. We can see that Bacterial category have the highest amount of frequencies, but if we look at their resulting probabilities, Stomatognathic is the winner and these words are more likely to appear in this category.

Probability of a word of an unknown sample is therefore calculated by the number of particular word occurences (from training samples) in the category divided by the total number of words (from training samples) in the category. These calculated probabilities are multiplied by each additional word in the unknown sample and form the resulting probability - **Likelihood**

The sample classification process contains one more multiplication with a probability called **Prior**. Since there is only one extra multiplication, in long-text samples these priors have a very low impact on resulting probability.

Just as we have taken into account the total number of words in the categories, we can also take into account the number of samples in each category. If we have category with rarer samples, we would like to reduce its probability allitle bit because of that.

The probability of Prior we get by the amount of samples in the category divided by total amount of samples in all categories. The more similar amounts of samples between categories, the less impact they will have on the result. However, even if there is a big difference between these amounts, the classification of long texts will still have little effect on the resulting probability.

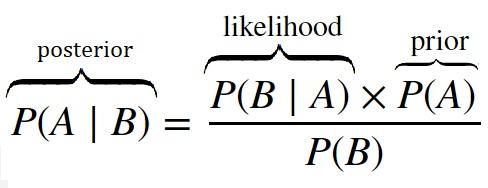
**Visualization of Prior impact**

Impact can be seen in the Probability of words graph. Because we have clicked on a sample with large amount of words, effect of prior is very weak and there is no visible difference in the resulting probabilities with and without priors that are marked with orange and purple colors.

In order to see impact of priors more clearly, we can reduce the text of the selected sample through “Words selection” slider which is right below. Select and pull the ending point somewhere to the beginning to pick for example just first two words. After that, we can see the resulting probabilities are differing much more. With each word added, we can see how these resulting probabilities are getting closer to each other.

**Bayes rule**

The process of calculating an unknown sample for specified category is inspired by already mentioned Bayes rule, where B is the unknown sample and A is a category



* More accurate expression of probability calculations is expressed in the web application.
* This algorithm can be modified to work with rational numbers in attributes (Gaussian Naive Bayes)
* The main disadvantage of this algorithm is the assumption of word independence - as we calculate probabilities of each word separately, it is assumed that these words arent in any relationship with each other
* The advantage of this algorithm is the speed of training and classification, it works well with small amount of training data and it can also handle large number of attributes as we could see in our case with text classification
* When calculating likelihood, there may be a problem with a certain words. If we have a category in the training dataset with certain zero word occurences, and this certain word is in the unknown sample, the probability of this word will result in zero, which would cause zero likelihood probability. Therefore, we need to use Laplace smoothing – adding one word occurrence to all category attributes (words)

