Wikipedia Summary Text Classificatiom

Maggie Faust

1 Introduction

In this project, I create a dataset for training a classifier to diferentiate between Wikipedia categories. The classifier sorts a wikipedia article into one of six categories based on its summary. I wanted to create something using Wikipedia because I like wikipedia and because I was already using the API for my capstone. I chose to make a classifier so I could reuse the code I made for Homework 2.

2 Dataset

In this section I will explain how the dataset was created. The dataset was created using the wikipediaapi extension. The dataset consists of words from the summaries of Wikepedia articles. The summary is the section at the beginning over the article. There are six categories that will be the classes for the classification model. The categories are "The arts," "Health," "History," "Science," "Religion," and "Technology."

Pages were sampled using a recursive method. The recursive method is called samplePages. At its most simplified, the way this method works is that it starts at one of the six categories and randomly samples a number of category members equal to the desired size of the sample. If the category member is an article, it is added to the final sample. If it is a category, we call the method on that category. This way, we sample pages that are members of the category, or one of its subcategories, or one of those categories' subcategories, etc. However, there are some more complications. The first complication is that, on Wikipedia, an article or category can be a member of more than one category. Because of this, the same article might be reachable from more than one of the six categories. From an NLP perspective, this means that articles can potentiallt belong to more than one of the classes. I was not able to fully acount for this but I was able to somewhat mitigate it. For each article or category I checked its supercategories, and those supercategories, and if any of those were one of the other five of the categories that are the classes it would be thrown out. I would have liked to checked more layers, but when I tried it drastically increased the time it took to generate the samples. Another thing I did to reduce the computation time was to add a maximum depth. The maximum depth was 4. When the maximum depth has been reached, if a category is sampled, it will sample another page from the current category instead of recursively calling SamplePages on the subcategory. This had the side effect that if it reached the maximum depth while in a category-only category, it would become stuck forever. Therefore, if it is about to go into the maximum depth, it checks if the next category is a category-only category, and ignores it if it is. Another problem is that it has to resample whenever a page that is neither an article or a category is picked. To reduce the time spent on pages that are neither articles or categories, I had it exclude categories which have "Category:Wikipedia images by subject" as a supercategory, so it was less likely to get stuck in a category that was majority images.

For each category, two samples were made, on that was a sample of ten pages, and another that was a sample of twenty pages. I will call them the 10-samples and the 20-samples respectively. The tokens from each pages' summary in a sample are stored in the same file, because the datset is designed to be used with models that use a bag-of-words approach. Punctuation marks like periods and commas are considered their own tokens for this dataset. For the dev set, twenty summaries were chosen from each class for the model to label.

3 Results

Table 1 shows the results of training two models on the training data. The two models are the ones I created for Homework 2, Naive Bayes and Logistic Regression. The models were trained with both

the dataset of 10-samples and the dataset of 20-samples. You can see that the Naive Bayes model actually performed better on the 10-samples than the 20-samples. For logistic Regression, it performed better on the 20-samples, but not by much. Since it isn't a very good model, i will be ignoring it for the rest of the report.

If we look at the structure of the samples, we can better understand this disrepency. The size of a Wikipedia pages summary can vary wildly. Some pages have a summary that is one sentence long, while others can have a summary that is multiple paragraphs long. Because of this, the token length of the samples is not consistant. You can see this by looking at Table 2, which shows the token counts of each sample. You can see in the table that, on average, the 20-samples had three times as many tokens as the 10-samples. Some categories have larger sample sizes than others. Therefore, this might mean that the model is more accurate for some classes than others.

In Table 3, the three most common keywords in each sample are shown. Punctuation and common words are ignored. It seems that, in general, the 10-samples top words contain more general words that correspond to to the topic of the corresponding category, while the 20-samples contain more specific vocabulary words that are only relevant to a few pages. For example the top words in the 10-sample for health are "health," "care," and "disease," whereas the top words in the 20-sample are "sleep," "tracking," and "may." I think the reason for this is that the 20-samples are more likely to contain a page with a significantly longer than average summary, which means that one page will be disproportionately represented in the sample, causing bias in the model. It is also possible that 20-samples are more likely to contain the same page twice, which would also cause a page to be over represented.

Table 3: Top keywords of each sample

	Health	History	Religion	Science	Technology	The Arts
10-Samples	• health • care • disease	• history • events • also	• religion • religious • prize	• science • lunar • factorization	• technology • entertainment • management	• art • music • arts
20-Samples	• sleep • tracking • may	• history • depp • heard	• mormons • summa • one	• science • scientific • web	• technology • society • scene	• art • social • artists

Continued on next page

Table 1: Metrics of different models on different samples (The values for Logistic Regression are averages)

	Naive Bayes on 10-Samples	Naive Bayes on 20-Samples	Logistic Regression on 10-Samples	Logistic Regression on 20-Samples
Accuracy	0.567	0.592	0.231	0.233
Precision	0.659	0.623	0.139	0.216
Recall	0.597	0.592	0.231	0.233
F1	0.609	0.607	0.152	0.207

Table 2: Token counts of samples

	Health	History	Religion	Science	Technology	The Arts	Average
10-Samples	1710	1979	1857	2221	1167	1445	1396.5
20-Samples	3101	2929	3492	3252	2947	3027	3124.667

4 Conclusion