Assignment 1

20/10/2016 - Due 03/11/2016 BIL 614 - Text Mining

Text Categorization with Vector Space Models

In this assignment you are expected to implement a document-term vector space, apply weighting functions and use these for text categorization. We have two important goals, first you will practice and probably realize what we mean by vector space models by implementing it. Second, you will have to devise a weighting function that will highlight dimensions (or words) discriminating the categories in the dataset.

Dataset

Dataset URL: `http://qwone.com/~jason/20Newsgroups/`

You will be using a dataset formed of about 20,000 news group messages collceted from 20 different groups. This dataset has the following groups:

```
comp.graphics
comp.os.ms-windows.misc
comp.sys.ibm.pc.hardware
comp.sys.mac.hardware
comp.windows.x
--------------------
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
--------------------
sci.crypt
sci.electronics
sci.med
sci.space
--------------------
misc.forsale
--------------------
talk.politics.misc
talk.politics.guns
talk.politics.mideast
--------------------
talk.religion.misc
alt.atheism
--------------------
soc.religion.christian
```

As you can see, while some groups are completely unrelated (e.g. alt-atheism and sci.crypt) some are very related to each other (e.g. talk.politics.mideast and talk.politics.misc).

In text categorization problems, the dataset is usually divided into two parts, the training set and the test set. Using the training set to form a model for the task, you will assign the documents in the test set to categories. You should download the training and test split by-date. This zip file has separated the dataset into two sets, using their date, which probably makes more sense as this is more similar to real-life applications of the dataset. You should download the file 20news-bydate.tar.gz.

An example from the dataset is given as below. Each file starts with a header, you can choose to remove these headers. Please keep in mind that if you choose to keep the header, the word ”From”, ”Subject” will appear in all messages, loosing its meaning (may even treat them as stop-words).

```
From: jvp4u@Virginia.EDU (Jeffery Vernon Parks)
Subject: Re: Info about New Age!
Organization: University of Virginia
Lines: 1

Suggestion: try "Exposing the New Age" by Douglas Groothuis.
```

Creating the Vector Space

You should first process the files with a tokenizer. You are free to use any tokenization techniques ranging from a simple whitespace split to lemmatization. While the results may vary depending on your choices in parsing the text files, I personally don’t expect big differences. You may use stemming, stop word removal or other techniques.

In the class we have discussed two vector space model implementations, the first is the hash table based quick & dirty approach. As the dataset is not big in size you can use this strategy as well. Also you can use sparse matrix representation with a vocabulary (as you can see from the dataset Website it is possible to use Octave/Matlab as well). You are free to use any programming language, but should not use any libraries except for general data structures, stemming or lemmatization. You should implement the vector space on your own.

As a text categorization method, you will use centroids of the vectors in the training set. So, for the training set you are expected to create 20 centroid vectors. You will experiment with two different weighting schemes. The first one is the generic tf-idf method. You can calculate the idf using the complete training set.

For the second one you should devise a weighting scheme which differentiates between words that appear in multiple categories and word that appear in just one category. You can check the article `http://www2009.eprints.org/21/1/p201.pdf` for inspiration.

Categorization and Results

Once you have created the vector space, you should compare the test documents’ vector with each centroid and assign the document to most similar category. You should think about normalization and weighting of the test and centroid vectors. In order to calculate the similarity you should use cosine similarity.

As you know the correct category for the test documents you can calculate Recall, Precision and F1 Measure for each category separately. In your final report you should create a table showing these three values for each category. Finally you should report average of the results (macro or micro average, just mention which one you are using and know the difference).

You have to submit:

- Your source code for the assignment
- A report containing
  - If you have done something different from white-space tokenization explanation of you procedure
  - Description of the weighting function you used for categorization
  - A table showing the results of your experiment
  - Any other discussion you find relevant
- A Readme file for instructions on compiling and running your code.

Submission

Your assignment is due for 03/11 Thursday 23:59. You have 3 late day submissions, with a penalty of 10% per day. Please try to include short and clear instructions for the code, a small readme file. You can submit your assignment to the web-page given below (you can update your submission as well), zip all the documents and upload.

`https://script.google.com/macros/s/AKfycbyP-qNzlBtVYTFlrysdXkhnMmrYikJVBRE5yyEOJMT4QqT03Snh/exec`

Enjoy...