Assignment 2 -

Document Classification, Clustering and Topic Models

In this assignments you will be using various algorithms for text classification, topic modeling and clustering in performing different types of text analysis. You will have to submit a comprehensive report and the accompanying code. The more thorough the report - the higher the grade.

Submission deadline: Dec 26, 23:59

Objectives:

1. Learn to work with the sklearn package.
2. Learn to design and interpret experiments in NLP (authorship recognition,classification, clustering, topic modeling).
3. Working with relatively big datasets.
4. Understand the differences between the various algorithmic frameworks and their application to different types of data.

Submission guidelines:

1. You should submit one zip file with all relevant code and reports. The file should be named <your id.zip>
2. The main requirement of this assignment is a report file explaining your use of the algorithms and describing the results - comparing results of different algorithms and different configurations of parameters. This file should be <your id>.pdf.
3. Code files should be well documented with a clear usage examples.
4. The input for the code files should be the appropriate raw input files provided below. Preprocessing should be done as part of the execution.
5. You should use Python 2.7 for the coding part.

1. Classification: Who Controls this Account

Politicians as well as other public figures usually have assistants and staffers that manage most of their social media presence. However, like many other norm defying actions, Donald Trump, the President of the United States is taking pride in his untamed use of twitter. At times during the presidential campaign it was hypothesized that Donald Trump is being kept away from his Twitter account in order to avoid unnecessary PR calamities. Trump's tweets are not explicitly labeled (Hilary Clinton, for example, used to sign tweets composed by her by an addition of '-H' at the end of the tweet while unsigned tweets were posted by her staffers). It is known, however, that Trump was using an android phone[1] while the staffers were most likely to use an iPhone. Luckily, the device information is part of the data available via the Twitter API, hence the device used can be used as an authorship label.

In this task you will be using a number of supervised machine learning classifiers in order to validate the hypothesis about Trump tweeting habits.

Algorithms:

You should use Python's nltk and sklearn packages for preprocessing, training and testing your classifiers (these packages are well documented and usage examples are part of the documentation).

You should use:

1. sklearn.linear_model.LogisticRegression
2. sklearn.svm.SVC (use both linear and nonlinear kernels!)
3. A third classifier of choice (from the sklearn package). You are encouraged to experiment with classifiers that allow combining different types of features (e.g. number of capitalized words, time of tweeting, etc.)

Make sure you do not test you code on data that was used for training. You are encouraged to use sklearn's cross validation module. Think about the evaluation measures you use.

Data:

A small dataset of a couple of thousands tweets from Trump's account posted between early 2015 and mid 2017 can be found in tweets.tsv, available to download at https://mega.nz/#F!3ZAVDByb!PxX4mJouWCRDbfmLGCVQbg
The file is tab separated, each tweet in a new line. The fields in the file correspond to:
<tweet id> <user handle> <tweet text> <time stamp> <device>

[1] Trump switched to a secured iPhone in April 2017, hence, building an accurate authorship model on older data can be used for authorship attribution of newer tweets.

While the data is already cleaned and filtered, there is still some degree of freedom you will have to take care of. Specifically:

1. The **handle** field: the handle field can take one of the following three user names: realDonaldTrump (this is Trump's account), POTUS (stands for President of the United States, this is the official presidential account, thus not Trump before the election) and PressSec - the official twitter account of the president's Press Secretary.
2. The **device** field: the device field can take various values ranging from 'android', iphone', instagram (will appear as 'Instagram'), a web client (will appear as 'Twitter Web Client') among other possibilities.
3. The format of the timestamp field is '%Y-%m-%d %H:%M:%S' you can use the *datetime* module and the *strftime()* and *strptime()* functions to parse and process time stamps.

You will have to be careful to use the right tweets and the right labels in training and testing.

Report

Your report should include a detailed list of model and data assumptions and should indicate the differences between the different algorithms and the various settings as well as a detailed comparison of the results. It should also include your insights and conclusions as learnt from the data. Specifically you should address the following:

1. What data was finally used - how it was preprocessed and filtered.
2. What is the data representation (input) for each algorithm.
3. What are the settings used for each algorithm.
4. Comparison between algorithms and settings.
5. If there are significant differences between algorithms/settings - why do you think that is.

Extra: can you verify the claim that Trump was kept away from his Twitter account during the campaign?

Topic Modeling and Clustering

Topic Models can be used for data exploration and as a first step in documents clustering. In this part of the assignment you will be applying Topic Models (LDA) and then cluster two very different datasets.
You should use Topic Models on the two datasets (separately) and report on the topics found. You should then use the topic assignments and cluster the data.

Algorithms

1. For Topic Modeling you should use sklearn.decomposition.LatentDirichletAllocation (note that this implementation uses EM and Variational Bayes approximation instead of Gibbs sampling. While faster, VB is a bit less accurate than Gibbs).
2. For clustering you should use sklearn.cluster.KMeans and sklearn.cluster.DBSCAN

Data

In this part of the assignment you will work on two very different datasets:

1. The first dataset is the same dataset of Trump's tweets you used for the first task.
2. The second dataset contains about three million comments submitted to the FCC regarding net neutrality.

The FCC dataset:

Background: the Federal Communication Commision is the federal agency in charge of regulating interstate communication channels such as radio, TV, cable and the internet. Citizens can sign petitions and post comments in support/against proposed regulations. Net Neutrality is a major regulatory issue that will be decided this December. It was recently claimed (recommended reading!) that many of the comments opposing net neutrality that were submitted to the FCC are not authentic and were submitted by bots that used simple linguistic manipulations in order to appear authentic. You will experiment with modeling and clustering of these data, trying to support/disprove these claims.

The dataset: The compressed dataset proc_17_108_unique_comments_text_dupe_count.csv.zip can be found here (~198MB). The uncompressed file is a .CSV file of 2.6GB (!). In case you are having storage/memory issues on your laptops/desktops you can contact Mendy Henner, the Department IT manager (mandy@bgu.ac.il) and he will allow server access for this assignment. If you contact him please CC me as well.

File format: The first line in the file is a header line defining the following fields:
'docid', 'text_data', 'dupe_count'

In order to read a .csv file properly you should use the csv package. This particular dataset is comma separated and textual fields are defined by the quote marks (“”).
A simple usage example (to count entries in the file) is:

```
with open('proc_17_108_unique_comments_text_dupe_count.csv', 'r') as
fin:
    c=0
    reader = csv.reader(fin, delimiter=',', quotechar='"')
```

```
    for row in reader: c+=1
print c
```

In order to understand the benefits of using the csv package compare the output of the code above the output of:
`wc -l proc_17_108_unique_comments_text_dupe_count.csv`

Report

Just like in the first part - you are required to submit a report (the same file) and the accompanying code. In your report you should address the following (as well as other issues you find relevant):

1. Are the topics discovered coherent? Do they (/some of them) make sense?
2. How did you choose the k (the number of topics)?
3. How did you choose c (the number of clusters)?
4. How are the topics and clusters change with different choices of k and c?
5. Do you see a significant difference in the performance on the different datasets? Can you speculate about this difference? Can this be solved heuristically?

Extra: use the clustering algorithms directly on the data, without the “dimensionality reduction” performed by the use of Topic Models. What’s the effect on runtime? On results?

Good Luck!