MIE 1624 Introduction to Data Science and Analytics – Fall 2020

Assignment 3

Due Date: 11:59 pm, November 23, 2020

Submit via Quercus

Background:

Sentiment Analysis is a branch of Natural Language Processing (NLP) that allows us to determine algorithmically whether a statement or document is "positive" or "negative".

Sentiment analysis is a technology of increasing importance in the modern society as it allows individuals and organizations to detect trends in public opinion by analyzing social media content. Keeping abreast of socio-political developments is especially important during periods of policy shifts such as election years, when both electoral candidates and companies can benefit from sentiment analysis by making appropriate changes to their campaigning and business strategies respectively.

The purpose of this assignment is to compute the sentiment of text information - in our case, tweets posted recently on US Presidential Elections 2020 - and answer the research question: ***"What can public opinion on Twitter tell us about the US political landscape in 2020?"*** The goal is to essentially use sentiment analysis on Twitter data to get insight into the 2020 American Elections.

Central to sentiment analysis are techniques first developed in text mining. Some of those techniques require a large collection of classified text data often divided into two types of data, a training data set and a testing data set. The training data set is further divided into data used solely for the purpose of building the model and data used for validating the model. The process of building a model is iterative, with the model being successively refined until an acceptable performance is achieved. The model is then used on the testing data in order to calculate its performance characteristics.

1) Produce a report in the form of an IPython notebook detailing the analysis you performed to answer the research question. Your analysis must include the following steps: data cleaning, exploratory analysis, model preparation, model implementation, and discussion. This is an open-ended problem: there are countless different ways to approach each part of the analysis and therefore the motivation for each step is just as important as its implementation. When writing the report, make sure to explain (for each step) what it is doing, why it is important, and the pros and cons of that approach.

2) Create 5 slides in PowerPoint and PDF describing the findings from exploratory analysis, model feature importance, model results and visualizations.

Two sets of data are used for this assignment. The *sentiemnt_analysis.csv* file contains tweets that have had their sentiments already analyzed and recorded as binary values 0 (negative) and 1 (positive). Each line is a single tweet, which may contain multiple sentences despite their brevity. The comma-separated fields of each line are:

0	ID	Tweet ID
1	text	the text of the tweet
2	label	the polarity of each tweet (0 = negative sentiment, 1 = positive sentiment)

The second data set, *US_Elections_2020.csv* contains a list of tweets regarding the 2020 US Presidential elections. The fields of each line are:

0	text	the text of the tweet
1	sentiment	1 for positive sentiment, 0 for negative sentiment
2	negative_reason	reason for negative tweets. NaN for positive tweets

Both datasets have been collected directly from the web, so they may contain html tags, hashtags, and user tags.

Learning objectives:

1. Implement functionality to parse and clean data according to given requirements.
2. Understand how exploring the data by creating visualizations leads to a deeper understanding of the data.
3. Learn about training and testing machine learning algorithms (logistic regression, k-NN, decision trees, random forest, XGBoost, etc).
4. Understand how to apply machine learning algorithms to the task of text classification.
5. Improve on skills and competencies required to collate and present domain specific, evidence-based insights.

To do:

1. **Data cleaning (20 marks):**
   The tweets, as given, are not in a form amenable to analysis – there is too much 'noise'. Therefore, the first step is to "clean" the data. Design a procedure that prepares the Twitter data for analysis by satisfying the requirements below.
   - All html tags and attributes (i.e., /<[^>]+>/) are removed.
   - Html character codes (i.e., &...;) are replaced with an ASCII equivalent.
   - All URLs are removed.
   - All characters in the text are in lowercase.
   - All stop words are removed. Be clear in what you consider as a stop word.
   - If a tweet is empty after pre-processing, it should be preserved as such.

2. **Exploratory analysis (15 marks):**
   - Design a simple procedure that determines the political party (Republican Party, Democratic Party and Others) of a given tweet and apply this procedure to all the tweets in the 2020 US elections dataset. A suggestion would be to look at relevant words and hashtags in the tweets that identify to certain political parties or candidates. What can you say about the distribution of the political affiliations of the tweets?
   - Present a graphical figure (e.g. chart, graph, histogram, boxplot, word cloud, etc) that visualizes some aspect of the generic tweets in *sentiment_analysis.csv* and another figure for the 2020 US election tweets. All graphs and plots should be readable and have all axes that are appropriately labelled.

3. **Model preparation (15 marks):**
   Split the generic tweets randomly into training data (70%) and test data (30%). Prepare the data to try multiple classification algorithms (logistic regression, k-NN, Naive Bayes, SVM, decision trees, ensembles (RF, XGBoost)), where each tweet is considered a single observation/example. In these models, the target variable is the sentiment value, which is either positive or negative. Try two different types of features, Bag of Words (word frequency) and TF-IDF. (*Hint: Be careful about when to split the dataset into training and testing set*)

4. **Model implementation and tuning (25 marks):**
   Train models on the training data and apply the model to the test data to obtain an accuracy value. Perform hyperparameter tuning and cross-validation, if necessary. Evaluate the same model with best performance on the 2020 US elections data. How well do your predictions match the sentiment labelled in the 2020 US elections data?

   Choose the model that has the best performance and visualize sentiment prediction results and the true sentiment for each of the two parties/candidates. Discuss whether NLP analytics based on tweets is useful for political parties during election campaigns.

Split the **negative** 2020 US elections tweets into training data (70%) and test data (30%). **Use true sentiment labels in the 2020 US elections data instead of your predictions from the previous part.** Choose three algorithms from classification algorithms (logistic regression, k-NN, Naive Bayes, SVM, decision trees, ensembles (RF, XGBoost)), train multi-class classification models to predict the reason for the negative tweets. Tune the hyperparameters and chose the model with best score to test your prediction reason for negative sentiment tweets. There are 5 different negative reasons labelled in the dataset.

Feel free to combine similar reasons into fewer categories as long as you justify your reasoning. You are free to define input features of your model using word frequency analysis or other techniques.

5. **Results (25 marks):**

   Answer the research question stated above based on the outputs of your first model. Describe the results of the analysis and discuss your interpretation of the results. Explain how each party is viewed in the public eye based on the sentiment value. For the second model, based on the model that worked best, provide a few reasons why your model may fail to predict the correct negative reasons. Back up your reasoning with examples from the test sets. For both models, suggest one way you can improve the accuracy of your models.

The order laid out here does not need to be strictly followed. Significant marks of each section are allocated to discussion. Use markdown cells as needed to explain your reasoning for the steps that you take.

Bonus:

We will give up to 10% bonus marks for innovative work going substantially beyond the minimal requirements. These marks can make up for marks lost in other sections of the assignment, but your overall mark for this assignment cannot exceed 100%. The obtainable bonus marks will depend on the complexity of the undertaking and are at the discretion of the marker. Importantly, your bonus work should not affect our ability to mark the main body of an assignment in any way. Any bonus work should be explicitly labelled as "Bonus" in its own section. You may decide to pursue any number of tasks of your own design related to this assignment, although you should consult with the TA before embarking on such exploration. Certainly, the rest of the assignment takes higher priority. Some ideas:

- Try word embeddings (https://en.wikipedia.org/wiki/Word_embedding) and N-grams as feature engineering techniques in addition to WF and TF-IDF.
- Explore Deep Learning algorithms and compare their performance to that of your best performing classification model.
- While the exploratory analysis section requires only two figures, you can explore the data further. You can also display the results of the model visually.

Tools:

- **Software**
  - **Python Version 3.X** is required for this assignment. Python Version 2.7 is not allowed.
  - Your code should run on the CognitiveClass Virtual Lab (Kernel 3).
  - All libraries and built-ins are allowed but here is a list of the major libraries you might consider: Numpy, Scipy, Scikit, Matplotlib, Pandas, NLTK.
  - No other tool or software besides Python **and its component libraries** can be used to touch the data files. For instance, using Microsoft Excel to clean the data is not allowed.
- **Required data files**
  - **sentiment_analysis.csv:** classified Twitter data containing a set of tweets which have been analyzed and scored for their sentiment
  - **US_elections_2020.csv:** Twitter data containing a set of tweets from 2020 on the US elections, which needs to be analyzed for this assignment
  - The data files cannot be altered by any means. The IPython Notebooks will be run using local versions of these data files.
- **Optional data files**
  - **corpus.txt:** corpus containing a set of words and their associated sentiment values
  - **stop_words.txt:** file containing an extensive list of stop words (could be beneficial for negative sentiments)
  - You may use these files if you wish but you are not required to.

What to submit:

1. Submit via Quercus portal a IPython notebook containing your implementation and motivation for all the steps of the analysis with the following naming convention:

   lastname_studentnumber_assignment3.ipynb

   Make sure that you **comment** your code appropriately and describe each step in sufficient detail. Respect the above convention when naming your file, making sure that all letters are lowercase and underscores are used as shown. **A program that cannot be evaluated because it varies from specifications will receive zero marks.**

2. Submit 5 slides in PowerPoint and PDF describing the findings from exploratory analysis, model feature importance, model results and visualizations. Use the following naming conventions **lastname_studentnumber_assignment3.pptx** and **lastname_studentnumber_assignment3.pdf**

Late submissions will receive a standard penalty:

- up to one hour late - no penalty
- one day late - 15% penalty
- two days late - 30% penalty
- three days late - 45% penalty
- more than three days late - 0 mark

Tips:

1. You have a lot of freedom with however you want to approach each step and with whatever library or function you want to use. As open-ended as the problem seems, the emphasis of the assignment is for you to be able to explain the reasoning behind every step.
2. While some suggestions have been made in certain steps to give you some direction, you are not required to follow them. Doing them, however, guarantees full marks if implemented and explained correctly.

TA:

Shimona Narang email: shimona.narang@mail.utoronto.ca