**CS 590 NLP**

**HW3**

**Language Models**

**Due 02/16 11:59 pm**

**Overall Goal:**

In this homework you will work with LMs to analyze a dataset.

|  |
| --- |
| **Useful python packages:** |
| For the LM, you may find the nltk package useful: <https://www.nltk.org/api/nltk.lm.html>  Or you can implement the LM manually if you prefer. (Following the n-gram model discussed in class.) |

|  |
| --- |
| **Dataset** |
| The dataset that will be used is the toxic comment dataset. This consists of toxic comments, so be cautioned when viewing the dataset.  You can find the training and test sets here: <https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data>  The training set (**train.csv)** will be used for training the LM, while the test set (**test.csv**) and labels **(test\_labels.csv)** will be used for testing and analyzing your models.  Take time to understand how the training set and test set are laid out. |

|  |
| --- |
| **Language Model Tasks** |
| You will create 2 required functions for LMs (you may create more functions for your own use, but you need to at least create these two as specified):  **train\_LM(path\_to\_train\_file)**  This method trains up to a bigram LM and returns the trained LM. **The format for the train file should follow the same format as the train data file!** The LM links have useful steps to help you train a LM in nltk.  **test\_LM(path\_to\_test\_file, LM\_model)**  This method tests an LM on a test file. That is, for each instance in the test file, it assigns a MLE score (refer to in class notes for how we approach this) for each test text. The function then generates an output file as the same format as the test file but adds a new column with the MLE score. **The format for the input file should follow that of the test file.**  Once these functions are implemented use them to accomplish the following tasks.   1. Create 3 LMs:    1. LM\_full -> trained on all the training data.    2. LM\_not -> trained on all the training data which toxic labels are 0.    3. LM\_toxic -> trained on all the training data which toxic labels are 1. 2. For each LM, test on: 1) the full test set, 2) the subset of data which is non toxic, 3) the subset of data which is labeled toxic 3. For each test file, make observations (e.g. averages of scores for LM\_full on the toxic subset compared to LM\_not) and analyze if your LMs seem to be successfully capturing the language they were trained on. 4. Write out the observations of LMs to a report document to hand in alongside your code. I will be looking for multiple comparisons of LMs across test sets. Additionally, any problems encountered or errors seen during creation of the LM/testing will help demonstrate your thoughtfulness in this homework. |

|  |
| --- |
| **Additional Tips/Guidance** |
| 1. The scores you achieve will not determine your final grade for this homework. Rather your thoughtfulness in approaching and analyzing your models. This means that lack in analysis will receive lower scores as indicated in the grading scale. 2. Note that preprocessing the text is up to you. This assignment is not specifically evaluating preprocessing like the first assignment, but here you have a chance to practice any preprocessing for your final project and future assignments. 3. Get familiar with the NLTK libraries on your own on small sets of data. Just copying and pasting the code without fully understanding it will end up being detrimental as you may not be able to accurately give proper analysis and observations of results.    1. Do not just copy and past the NLTK class code. You may use portions from the nltk classes, but too much unused code from classes will result in lost points due to poorly written code. Experiment with the code and only import what you need. 4. DO NOT SHARE CODE. I have linked NLTK libraries which will be useful for this assignment. You may also come to me to discuss/figure things out. 5. Start the assignment early. You will have 2 weeks to complete this assignment, which is plenty of time if you start early any familiarize yourself with the libraries. **IF YOU WAIT UNTIL THE LAST MINUTE TO START, YOU WILL BE LESS LIKELY TO DO WELL.** I won’t be granting any additional extensions so each late assignment with follow the late grade policy. 6. Ask questions/approach like a researcher. Think like this is your chance to explore NLP models and analyze their effectiveness. |

**Grading**

Assignment will be graded as follows:

|  |  |
| --- | --- |
| **Description** | **Points** |
| Code Runs | 5 |
| LM Implementation/Tasks | 20 |
| Report analysis and observations | 20 |
| Documentation (Comments, functions, etc) | 5 |
| **Total:** | **50** |