Paul Boschert
09/18/2015
CSCI 5622 - Machine Learning: Feature Engineering (HW 3) Analysis

1. In order to test the accuracy of my training data set, I first used sklearn.metrics.accuracy_score. This gave me the accuracy of my model based on the y values in the training set. By itself, this is not very useful.
   (a) My training data set was clearly overfitting at ∼99%.
   (b) For a baseline, I submitted my score to Kaggle and recieved a poor score ∼63%.
2. So the next thing I did was add the ability to split up my data set for testing purposes so that I could train on part of it and use the other part for cross validation.
   (a) This is enabled with the --split flag.
   (b) I tried several different ways to split it up and settled on a 75% train, 25% test split.
   (c) My training score decreased slightly, and now I had a cross validation score of ∼63%
   (d) I resubmitted to Kaggle (because the submission timer was about to expire), and my score didn't change much.
3. Next, I tried adding the page listed in the spoiler file to the list of features.
   (a) This is enabled with the --page flag.
   (b) My training score remained about the same, and now I had a cross validation score of ∼67%
   (c) I also noticed that several of the pages were showing up in the top list of features:
   Pos: spartacusbloodandsand prisonbreak sherlock thewalkingdead gokai criminalminds fringe twentyfour torchwood-miracleday americanhorrorstory
   Neg: wings boymeetsworld mythbusters video skills cornergas tv onethousandwaystodie frequently theitcrowd
   (d) I did not submit this alone to Kaggle.
4. Next, I tried adding the trope listed in the spoiler file to the list of features.
   (a) This is enabled with the --trope flag.
   (b) My training score remained about the same, and now I had a cross validation score of ∼72%
   (c) When I submitted this to Kaggle, my score still didn't change much even though my cross validation set was significantly higher. After looking at the discussion board, I realized that the tropes and page in the training set were not in the test data set.
5. At this point I wanted to group the pages and/or the tropes together into some abstract feature.
6. I read a bit of documentation online and found a few papers discussing the basis of this assignment [1] [2]
7. At this point I tried adding the first genre listed for a page by quering the Open Movie Database. Then I appended the word 'genre' to the genre (e.g. 'genredrama')
   (a) This is enabled with the --genre flag.
8. I also tried adding the average decade in which the show ran by querying the Open Movie Database, and converting the years it was running to an average, then appending 'year' to the number (e.g. 'year2012')
   (a) This is enabled with the --year flag.
9. These two additional features did not result in a substantial improvement. I also removed trope and page since they were not in common with the testing data set. However, my Kaggle score was much closer to my validation set.
10. In addition, several genres were in the top features.
11. Instead of adding the year, I averaged the years listed, and rounded to the nearest decade.
    (a) This is enabled with the --decade flag. And it overrides the --year flag.
12. I also tried using a lemmatizer and a stemmer to simplify the words in the sentance. Either of these and in combination worsened my score on the validation data set and my Kaggle score so I removed them.
13. I tried changing ngrams as an option into the CountVectorizer as well, I settled on ngrams 1 through 2. 3 or even 4 made my validation data set worse consistently.
14. At this point I had a validation data set score and a Kaggle score that was doing pretty well, low ∼70%s
15. I found that I could increase my validation data set score up to 77%, but this did not translate to a Kaggle score of 77%. I surmised that this had something to do with how my features were too tied to the training data set.
16. The last thing I tried was reducing the number of features selected with sklearn.feature_selection.SelectKBest. I used the chi2 score_func in SelectKBest which selects the χ^2 stats of non-negative features for classification tasks.
    (a) This is enabled with the --featsel flag.
17. I also added the ability to change the feature selection ratio with the --featsel_ratio flag. I found the best value to be .9.
18. This finally gave me my best Kaggle score of 76.829%
19. Unfortunately, the results on Kaggle were only one half of the data set and my score droped to 74.560% on the other half of the data set... I suppose that's always possible when the algorithm is stochastic.

References

[1] Jordan Boyd-Graber, Kimberly Glasgow, Jackie Sauter Zajac *Spoiler Alert: Machine Learning Approaches to Detect Social Media Posts with Revelatory Information, https://github.com/ezubaric/jbg-web/blob/master/docs/2013_spoiler.pdf*, 2013. PDF file.

[2] Shawn M. Jones, Michael L. Nelson *Avoiding Spoilers in Fan Wikis of Episodic Fiction, http://arxiv.org/pdf/1506.06279*, 2015. PDF file.