**CS 505 – Spring 2022 – Assignment 2 (100 pts, bonus: 10 pts) – Text Classification**

**Problems due 11:59PM EST, March 7, 2022.**

---

In this assignment, you will learn about **text classification**, and use python libraries such as **sklearn** and **gensim**, which are popular in NLP. You have slightly less than 2 weeks to finish this particular assignment.

Submit in Gradescope by 11:59PM EST, March **7**, 2022
–Download the submission (answer sheet) template at this **link** and use it to write down your write-up answers.
–Please indicate names of those you collaborate with.
–Every late day will reduce your score by 20
–After 2 days (i.e., if you submit on the 3rd day after due date), it will be marked 0.

**Submit your submission (answer sheet) template complete with link to your code (Jupyter Notebook)**

**When necessary, you must show how you derive your answer.**

**Problem 1.** *Naive Bayes and Logistic Regression Classification (40 pts)*

1. *Consider the task of learning **Y** from **X**, where the class label **Y** $\in 0, 1$ and each input **X** is represented with $n$ features i.e. **X** $= \langle X_1, X_2, ..., X_n \rangle$, where each $X_i$ is a continuous variable that follows a Gaussian distribution.*
   (a) *(3 pts) List the parameters that you would need to learn to classify an example using a Naive Bayes (NB) classier – you need to list what the parameters are, not just how many they are.*
   (b) *(2 pts) What is the total number of parameters you need (in terms of $n$)?*
2. *Consider a simple classification problem using Naive Bayes to determine whether a review of a beach resort is positive (1) or negative (0) i.e., $Y : Sentiment \in 0, 1$ given two features: (1) whether the review contains mention of the summer season $X_1 : Summer \in 0, 1$ and (2) whether the review contains mention of the rowdiness of the resort $X_2 : Rowdy \in 0, 1$. From the training data, we estimate that $P(Sentiment = 1) = 0.6$, $P(Summer = 1|Sentiment = 1) = 0.9$, $P(Rowdy = 1|Sentiment = 1) = 0.3$, $P(Summer = 1|Sentiment = 0) = 0.4$, and $P(Rowdy = 1|Sentiment = 0) = 0.7$. Assume that the data satisfies Naive Bayes assumption of conditional independence of the feature variables given the sentiment label.*
   (a) *(2 pts) Write down Naive Bayes formulation of $P(Y|X)$ using both the features $Summer$ and $Rowdy$ and the decision rule using $P(Y|X)$ i.e., how do you decide using $P(Y|X)$ if a review is positive or negative?*
   (b) *(10 pts) What is the expected error rate of your Naive Bayes classifier? i.e., the probability of observations where the label is different than predicted.*
   (c) *(2 pts) What is the joint probability that the sentiment of a review is positive and that the review contains mentions of $Summer$ and $Rowdy$ i.e., $P(Sentiment = 1, Summer = 1, Rowdy = 1)$?*
   (d) *(3 pts) Your boss decides to add another feature to your Naive Bayes classification model that is whether or not the review contains mentions of the Winter season $X_3 : Winter \in 0, 1$. Assume that a review that contains mentions of season can mention either $Summer$ **or** $Winter$ but cannot mention both i.e., since $Summer$ == $\neg Winter$ it cannot have $Summer = 1$ **and** $Winter = 1$ (and similarly, it cannot have $Summer = 0$ **and** $Winter = 0$). In this case, are any of the NB assumptions violated? Why? What is the joint probability that the sentiment of a review is negative and that the review contains mentions of $Winter$ and $Rowdy$ and does not contain mention of $Summer$? i.e., $P(Sentiment = 0, Summer = 0, Rowdy = 1, Winter = 1)$?*
   (e) *(8 pts) What is the expected error rate of your NB classifier using these three features?*
   (f) *(3 pts) Does the performance of your NB classifier improve with this addition of new feature $Winter$? Explain why.*
3. *Imagine that a certain important feature is never observed in the training data e.g., mentions of cleanliness $Clean \in 0, 1$, but it occurs in the test data.*
   (a) *(2 pts) What will happen when your NB classifier predicts the probability of this test instance? Explain why this situation is undesirable.*
   (b) *(5 pts) Will logistic regression have a similar problem? Explain concretely why or why not by looking at the formulation of the weight update in logistic regression.*

**Problem 2.** *Twitter Sentiment Classification with sklearn (40 pts, bonus: 10 pts)*

*The file: sentiment-train.csv contains 60k tweets annotated by their sentiments (0: negative, 1: positive), which is a sample of a very large sentiment corpus that has been weakly annotated based on the emojis contained in the tweets. See here for the full description of the data and to download the full corpus (Note that the full corpus contains "neutral" tweets, which we do not include in our test set: sentiment-test.csv).*

1. *(5 pts) Using sklearn, train a Multinomial Naive Bayes classifier (with default parameters) to predict sentiment on the training data, featurizing the data using CountVectorizer (also in sklearn). Use the default parameters of CountVectorizer and* ***max_features = 1000*** *(to limit the number of bag-of-word features to only the top 1k words based on frequency across the corpus). You should learn more about CountVectorizer parameters and usage here. Report the accuracy of the trained classifier on the test set.*
2. *(3 pts) Use CountVectorizer with binary counts (set binary flag = True), with other parameters same as before. Using these features, train MultinomialNB classifier with default parameters and report the accuracy of the trained classifier on the test set. Does using binary counts as features improve the classification accuracy?*
3. *(5 pts) Using sklearn, train a logistic regression classifier on your training data, using CountVectorizer to featurize your data (with the same parameters as in part 1). Report the accuracy of the trained classifier on the test set. Which classifier performs better on the test set?*
4. *(2 pts) Train a logistic regression classifier as before, using* ***binary*** *CountVectorizer to featurize your data. Report the accuracy of the trained classifier on the test set.*
5. *(2 pts) After performing the above experiments, which feature extractor and statistical model combination is good for your dataset? Note that this step is called model selection. Read online about the following terminology "model selection" and "development set" a.k.a. "validation set" and describe if it is okay to do model selection on the test set.*
6. *Use StratifiedKFold in sklearn to split your training data into 10 splits while maintaining label proportions of your training data.*
   (a) *(8 pts) Conduct 10-fold cross validation experiments on your training data: training a Multinomial NB classifier with CountVectorizer and different max_features (= 1000, 2000, 3000, or 4000) with and without binary counts. Report the average accuracies of these different max_features and binary/not binary across folds.*
   (b) *(2 pts) Select the combination of max_features value and binary/not binary count choice that has the highest average accuracy in your cross-validation experiments and train a Multinomial NB classifier on your* ***whole*** *training data using this parameter to featurize your data. Report the accuracy of this trained classifier on the* ***test*** *set.*
7. *Using word2vec as dense features for classification*
   (a) *Use gensim library to learn 300-dimensional word2vec representations from the tokenized tweets (you can use Spacy for tokenizing tweets) in your* ***training*** *data (you can use default parameters).*
   (b) *Given the learned word2vec representations, construct a vector representation of each tweet as the average of all the word vectors in the tweet. Ignore words that do not have vector representations – since by default gensim word2vec model only learns vector representations for words that appear at least 5 times across the train set.*
   (c) *(7 pts) Train a logistic regression classifier using the above vector representation of tweets as your features. Report the accuracy of the trained classifier on the test set. Does dense feature representation improve the accuracy of your logistic regression classifier?*
8. *For the discussion questions below, please be concise and make sure you briefly answer each point.*
   (a) *(3 pts) Research for and describe one additional statistical model which you could use to fit your data. Why do you think this model is appropriate to be used with your text features? Briefly compare this model to logistic regression.*
   (b) *(3 pts) Class imbalance is a frequent problem in many ML applications where some classes are more frequent than others. Describe at least one way of addressing it in one of the methods you have considered so far.*
9. *(Bonus: 10 pts). Train a Multinomial NB classifier with CountVectorizer (with the best number of max_features and binary/non binary count via cross-validation like 2.6 above) on the entirety of the 1.6m twitter sentiment training data that you can download from here. To help reduce the burden of processing on CountVectorizer, you can first tokenize each tweet using Spacy. Report the accuracy of this trained classifier on the* ***test*** *set. Does having a huge amount of training data allow this simple classifier like NB with this simple bag-of-words features to perform even better on the test set?*

**Problem 3.** *Explainability in NLP (20 pts)*

1. *Logistic Regression*

   (a) *(1 pt) In 2.1 you used logistic regression where your features were from CountVectorizer. What do you think each of 1000 features you created represent and what do they correspond to? You may find* `get_feature_names_out()` *method for CountVectorizer helpful.*

   (b) *(3 pts) Examine the coefficients in your trained logistic regression model. Can you find the coefficients that correspond to the word "good" and "bad"? How do they compare to each other and in which way (increasing or decreasing the probability of positive) they contribute to the prediction?*

2. *Gradient Tree Boosting is another powerful statistical tool to learn from data. It is an ensemble model meaning it is a collection of other smaller models i.e. decision trees.*

   (a) *(2 pt) After reading about gradient tree boosting, how would you explain the method to your boss who is technical but non-expert?*

   (b) *(4 pts) Again using your features obtained from CountVectorizer on the 60k sentiment data, use one of the following libraries to fit a model: catboost, xgboost, lightgbm. All three libraries have nice tutorials you can follow. You may use default parameters. Provide accuracy on your train and test sets.*

   (c) *(3 pts) Now, we are interested in again which feature (word in the CountVectorizer case) contributed greatest to the prediction. SHAP values are great in interpreting how the features contribute to the predictions made by complicated models. Read about SHAP values online and describe how would you explain them to a colleague at work? Be thorough, yet concise.*

   (d) *(4 pts) Create a beeswarm plot using the shap library for Python (*`shap.plots.beeswarm`*). Describe what the axis, colors, legend and dots mean. Give examples to words which lead to positive and negative sentiment predictions. You may find a sample figure in the resources pack.*

   (e) *(3 pts) Using* `shap.plots.force`*, examine one interesting example, print the actual tweet and discuss what the plot tells you and identify the significant words that led to the score. What does the bold number indicate? You may find a sample figure in the resources pack. (Hint: Examine the documentation for the function, and try filling in the first four arguments:* `base_value`*,* `shap_values`*,* `features` *and* `feature_names`*. For feature names, you may use your CountVectorizer as in 1a.)*