Final Project report

Naman Rastogi

U1472278

An overview of the project.

This project was a classification problem which had binary results, 0 and 1. 0 for guilty and 1 for not guilty. We had 4 parts of the dataset: bow, tfidf, glove and misc. We had the train data, test data and eval data. Eval data is the data for which we have to make predictions.

What are the important ideas you explored?

These are my main 6 submissions.

1. Simple perceptron with a bow.
2. Enhanced perceptron with bow.
3. Support vector machines(SVM) with tfidf and misc data.
4. Logistic regression with tfidf and misc data.
5. Simple perceptron with Glove.
6. Adaboost with a perceptron on tfidf and misc data.

One of the main things I learned and did for this project was preprocessing. I had to convert the labels from 0 and 1 to -1 and 1. This was needed so that my simple perceptron, support vector machines and logistic regression can work.

Another important thing I did was adding the misc dataset. The misc dataset had categorical values therefore it had to be converted to numerical form like all the remaining features. To do this, I used label encoding. It also had missing values for a column named as age, so I managed the missing values using the median.

Then, as misc dataset didn't have labels, they had to be combined with an existing dataset. Therefore I combined them with bow, tfdif and glove datasets.

Another interesting thing I used was the sklearn library. I used sklearn to improve my simple perceptron. I also used sklearn for label encoding.

I also used different hyperparameters to get the best result.

What ideas from the class did you use?

I used algorithms like Simple perceptron, Support vector machines(SVM), logistic regression(LR) and adaboost on simple perceptrons.

I also used concepts like hyper parameter tuning, ensemble learning, evaluating the models based on accuracy, cross validation, splitting the training data.

I used other models like decision trees, left them as they took a lot of time to train. I left the aggressive perceptron on tfidf as it gave a kaggle score of 0.35. Same for aggressive perceptron on bow.

The only thing different from the class that I used was the sklearn library.

What did you learn?

I learned how to use the concepts on real world data. I had some difficulties. A better way would be to tell these would be by stating the challenges.

1. Since perceptron was coded to handle the labels 1 and -1 and the data had the labels 0 and 1, I was getting 0.50 accuracy. It took me a while to figure out exactly how this was happening. So after that I did improve my accuracy.
2. We were using a development set and using its accuracy for the assignment. This was making my perceptron very very slow. So I had to remove it.
3. Misc dataset was an incomplete dataset as it didn't have the label attribute and it also had missing values. The missing values were fixed by using median but the variables this dataset had were categorical. So earlier I did one hot encoding but that led to dimensionality issues. So after that I did label encoding which combined the misc dataset with all 3 other datasets.
4. Accuracy was a big issue in this project. Even for some models where test accuracy was good/high the kaggle score was very bad. Example: Test accuracy for the adaboost with simple perceptron on bow was 0.65 but the kaggle score was 0.30.
5. Enhancing the result using hyperparameter tuning.
6. Altering the code like using batch size to make the code runnable in a shorter time.
7. A very interesting observation was that after applying adaboost on a simple perceptron, the kaggle score was 0.49. Without adaboost, simple

perceptron gave better results. For bow it gave 0.67 whereas with adaboost it gave 0.30.

A summary and discussion of results

Let's dive deep into the submission:

1. Simple Perceptron with bow: This was the simple perceptron which we used in the assignment. It gave me a test accuracy of 0.634 and the kaggle score of 0.67. I tried this with tfidf which has a test accuracy of 0.66 but the kaggle score was 0.34.
2. Enhanced perceptron with bow: Perceptron which used sklearn. It gave the kaggle score of 0.65.
3. Support vector machines with tfidf+misc: Used SVM which we used in our assignment. learning_rates = [0.1, 0.01, 0.001], Cs = [0.01, 0.1, 1] , epochs = 50. Batch size chosen as 100. Kaggle score of 0.79.
4. Logistic regression with tfidf+misc: Used logistic regression which we used in our assignment. learning_rates = [0.1, 0.01, 0.001], sigma2_values = [0.01, 0.1, 1], epochs = 100, batch_size = 64. Kaggle score of 0.79.
5. Simple perceptron with glove dataset: Used simple perceptron with glove dataset. Kaggle score: 0.55.
6. Adaboost on simple perceptron on tfidf and misc: Used ensemble learning of simple perceptron and adaboost. Kaggle score: 0.49.

If you had much more time, how would you continue the project?

I would try to find better ways to improve accuracy. My models were working great on the assignments but performed very badly on this data.

Try to train models quicker like decision trees.

Use random forests, if decision trees can be trained quickly.

Use more complex machine learning/deep learning models.