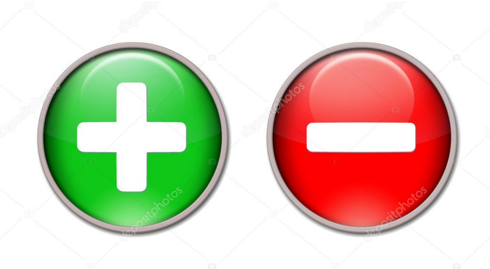
Semantic Analysis of Kaggle Dataset for Amazon Reviews

CSCI 561 - Homework 2

Shamim Samadi

Goal: perform semantic analysis on Amazon reviews in Python

- Determination of whether a review is "**positive**" or "**negative**"
- binary classification problem



Steps:

1. **Pre-processing:**
   - Bag-of-words representation
   - tokenization, remove punctuation, remove the stop words, lower case
   - train-test split: every 5th sample belongs to test, the rest is training data
2. **Classifier training**
   - Naive Bayes
   - Decision Tree
   - Neural Networks (multilayer perceptron)
3. **Classifier evaluation on test data**
   - F-1 score, precision, recall
   - ROC curve, AUC score

Naive Bayes was chosen as the 3rd classifier because:

- ➢ model is simple and computationally efficient
- ➢ a popular choice for text classification applications
- ➢ the feature independence assumption seems to go well with the bag-of-word representation

Results & Discussion:

Precision

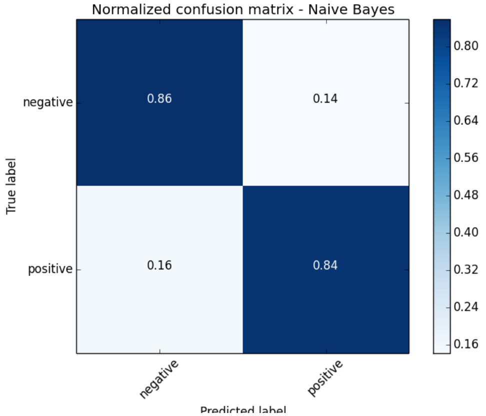
	class 0 (negative)	class 1 (positive)	average/total
Naive Bayes	0.84	**0.86**	0.85
Decision Tree	0.76	0.77	0.77
Neural Network	**0.85**	**0.86**	**0.86**

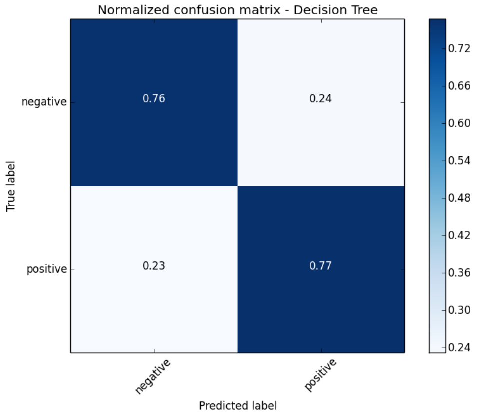
Recall

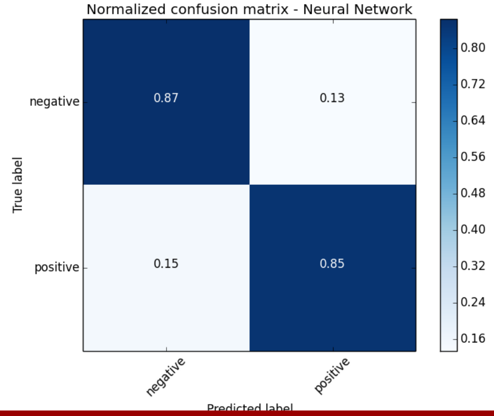
	class 0 (negative)	class 1 (positive)	average/total
Naive Bayes	0.86	0.84	0.85
Decision Tree	0.76	0.77	0.77
Neural Network	**0.87**	**0.85**	**0.86**

F1-Score

	class 0 (negative)	class 1 (positive)	average/total
Naive Bayes	0.85	0.85	0.85
Decision Tree	0.76	0.77	0.77
Neural Network	**0.86**	**0.86**	**0.86**


Normalized confusion matrix - Naive Bayes


Normalized confusion matrix - Decision Tree

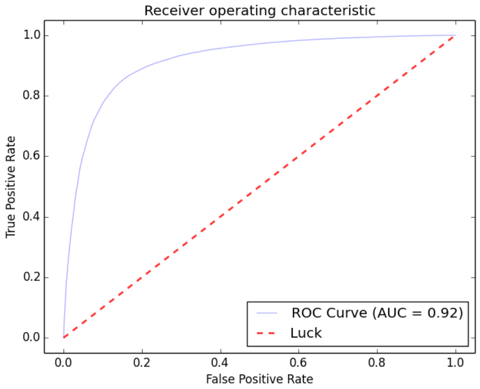

Normalized confusion matrix - Neural Network

Results & Discussion:

- Performance ranking
  1. Neural Network
  2. Naive Bayes
  3. Decision Tree

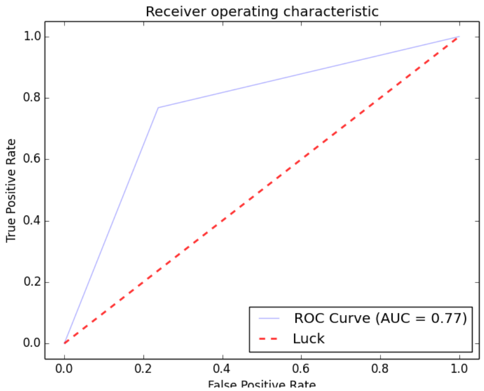
- Although neural network classifier has a slight edge in performance when it comes to evaluation metrics, the Naive Bayes classifier would be an overall better choice since neural network requires intensive training and is computationally very expensive.
- Decision tree performs poorly in high-dimensional feature spaces (which is the case in the bag-of-words feature representation used in this homework)
  - ➢ tricky to find the “key” tokens (as the key nodes) to split the tree on

Naïve Bayes



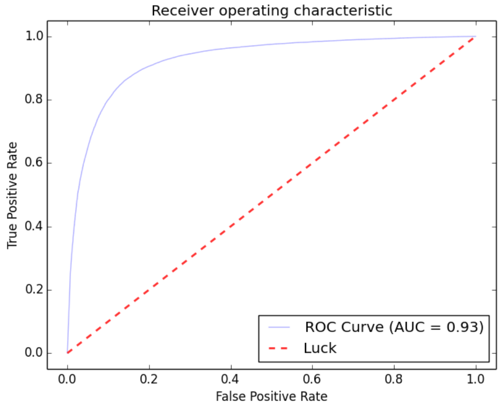
AUC = 0.92

Decision Tree



AUC = 0.77

Neural Network



AUC = 0.93