CS671A - Introduction to Natural Language Processing
Assignment 2 - Report

Rohit Gupta

150594

Contents of zip file

- Report

- Codes :
  bow_tfidf.py classifiers.py glove.py glove_wgt.py rnn.py word2vec.py word2vec_wgt.py

- Results :
  results_glove.txt results_w2v.txt results_bow.txt results_rnn.txt

Code Structure

1. **bow_tfidf.py** loads the given dataset and converts the documents into Binary bag of words representation, normalized Term frequency (tf) representation and Tfidf representation. We save the representations in sparse matrices (.npz format) to be used again. **results_bow.txt** contains the accuracy reported by various binary classifiers.
2. **glove.py** and **word2vec.py** store the documents in averaged word2vec format using pre trained word vector models. **glove_wgt.py** and **word2vec_wgt.py** do the same using tfidf values as weights for words. **classifiers.py** is used to run the classifiers. **results_glove.txt** and **results_w2v.txt** store respective results.
3. **rnn.py** trains and tests RNN(using LSTM) on word2vec and glove representations (both weighted and non-weighted) and reports accuracies. Results are in **results_rnn.txt**.

Details about dataset

1.	Main Dataset	- Stanford imdb movie review dataset (v1)
2.	Word2vec	- Google-news-vectors-negtive300
3.	Glove	- glove.6B.300d.txt (Stanford)

Number of training points = 25000
Number of testing points = 25000

CS671A - Introduction to Natural Language Processing Assignment 2 - Report

Rohit Gupta

150594

Results

Accuracies for different classifiers and representations are reported in the following table :

Classifier ↠ **Representation↡**	Linear SVM	MLP Classifier	Naive-Bayes Classifier	Logistic Regression	RNN (LSTM)
Binary BOW	85.10%	85.99%	82.60%	87.06%	---
Normalized tf	84.54%	86.40%	82.60%	80.61%	---
Tfidf	87.86%	85.47%	82.60%	88.80%	---
Average word2vec	85.64%	85.65%	74.94%	85.19%	85.65%
Weighted Average word2vec	83.56%	83.42%	74.24%	83.22%	83.42%
Average glove	83.29%	83.13%	72.88%	83.13%	83.86%
Weighted Average glove	80.43%	80.51%	71.48%	80.46%	80.85%

NOTE :

1. Naive-Bayes Bernoulli was used in Naive-Bayes Classifier.
2. Time-steps = 1 in case of RNN.