Assignment 3

CPSC 599/601: Practical Natural Language Processing

February 22, 2024

This Assignment will evaluate your skills in topic modeling and classification. Include in your submission the code used to generate answers as a Jupyter Notebook or Python program file as well as any files generated. Make sure it is clear what code answers each question.

This Assignment is meant to be completed individually. You may discuss the questions at a high level with other students but the final work submitted must be your own. Please reference any external resources you use to complete this Assignment using ACM referencing format.

Exercise 1 (20 marks)

A) Use the 20 Newsgroup dataset (from sklearn.datasets import fetch_20newsgroups) and the LDA (Latent Dirichlet Allocation) topic modeling algorithm from Gensim to identify the top 10 most common topics in the dataset. Explain the preprocessing steps you applied (and why). (**10 marks**)

B) Coherence scores are one way to decide on the number of topics to consider from your topic model. Use CoherenceModel and get_coherence() to obtain coherence values from LDA (use c_v). Plot the number of topics (select a range to explore) vs coherence scores using matplotlib. Did LDA perform well in identifying topics? From coherence, how many topics are there in the dataset? (**8 marks**)

C) How are the topics you selected as relevant based on coherence distributed among the documents? (**2 mark**)

Exercise 2 (20 marks)

Assume you have been given a dataset of 5000 observations, with 80% of the data labeled as "positive" and the remaining 20% labeled as "negative". Below is example code that can be used to generate a dataset for testing purposes.

```python
from sklearn.datasets import make_classification

# Generate a synthetic dataset with 5000 observations and 4 features
# with 80% positive and 20% negative labels
X, y = make_classification(n_samples=5000, n_features=4,
                           n_informative=4, n_redundant=0,
                           n_repeated=0, n_classes=2,
                           class_sep=2, weights=[0.8, 0.2],
                           random_state=1)
```

1. Explain why stratified data splitting could be important in this scenario and show one way it can be implemented in Python with the generated dataset. (**4 marks**)
2. Show a way in Python to balance out the dataset using over- or undersampling. Prepare the data for simple hold-out validation. (**4 marks**)
3. Prepare the data for k-fold cross-validation with 3 folds. Train a Naive Bayes classifier with this dataset. (**8 marks**)
4. Explain the concept of nested cross-validation and how it can be used to optimize hyperparameters in a machine learning model. Explain what the risk of using another cross-validation technique, say k-fold cross-validation when tuning hyperparameters. (**4 marks**)

Hint in text book exercise

Exercise 3 (20 marks)

For this exercise, you will use a dataset from the UCI Data Repository [1] with sentences from 3 different companies labelled with positive or negative sentiment. It can be downloaded here.

A) Train a Random Forest model to classify positive and negative sentiment. Report Precision, Recall, and F1 for each category (e.g., Pos and Neg) for your model. (**12 marks**)

B) Discuss your results from Part A. How could the addition of an extra class affect the learning algorithms? Did you need to account for any particular variables/characteristics in the dataset when training your model? Why or why not? (**6 marks**)

C) Would it be a good idea to perform 10-fold cross validation with this dataset? Why or why not? (**2 marks**)

Exercise 4 (10 marks)

A) Write a Python program to load the ELECTRA model for classification and provide a summary of the model architecture, including the number of layers, number of parameters, and layer types. (**5 marks**) Similar pre-train ex notebook => summary

B) Explain the concept of "freezing" layers in a neural network, and why it can be useful when fine-tuning a pre-trained model. Freeze all but the classification layers of the model in Part A. (**3 marks**)

C) Explain the steps involved in fine-tuning a pre-trained model for a specific NLP classification task. How does the process differ from training a model from scratch? (**2 marks**)

References

[1] Dimitrios Kotzias, Misha Denil, Nando de Freitas, and Padhraic Smyth. 2015. From Group to Individual Labels Using Deep Features. *In Proceedings of the 21th ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining (KDD '15). Association for Computing Machinery, New York, NY, USA, 597–606. https://doi.org/10.1145/2783258.2783380