Homework 4: Text Models & Neural Networks

New Attempt

- Due Nov 26, 2024 by 6:10pm
- Points 22
- Submitting a file upload

Instructions: *Please submit your answers as 2 files uploaded to Courseworks: a Jupyter Notebook (.ipynb) file & a pdf export. Please double check that all pages exported properly, sometimes they get cut off! In answering each of the following questions please include (a) the question as a markdown header in your Jupyter notebook, (b) the raw code that you used to generate any results, tables, or figures, and (c) the top ten or fewer rows of the dataframe (do not include more than ten rows for any table in your report). Include any plots or figures generated from your code as well.*

Part A: Build a classification model using text data

For Part A, you will be solving a text classification task. The training data is stored in the Homework 4 Data folder. The data consists of headlines that have been labeled for whether they are clickbait.

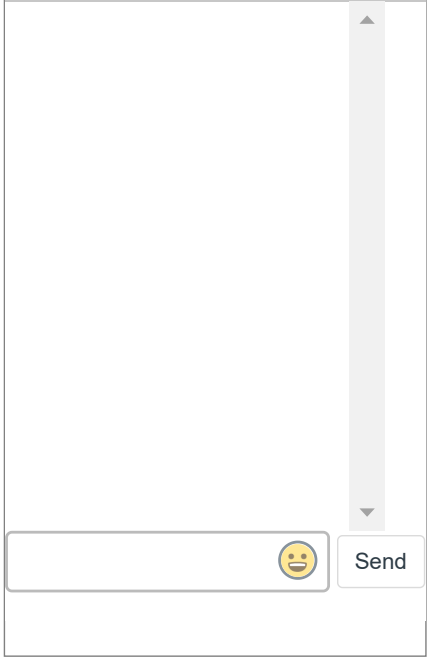
1. Import the data. The headlines will become your vectorized X matrix, and the labels indicate a binary classification (clickbait or not).
2. Convert the headline data into an X feature matrix using a simple bag of words approach.
3. Run logistic regression to predict clickbait headlines. Remember to train_test_split your data and use GridSearchCV to find the best value of C. You should evaluate your data with F1 scoring.
4. Run 2 more logistic regression models by changing the vectorization approach (e.g. using n-grams, stop_words, and other techniques we discussed). In both cases, keep your logistic regression step the same. Only change how you're generating the X matrix from the text data.
5. Which of your 3 models performed best? What are the most significant coefficients in each, and how do they compare?

Part B: Build a Predictive Neural Network Using Keras

In Part B, you will run a multilayer perceptron on the iris dataset to predict flower type.

1. Load the data. Data can be imported directly using pd.read_csv() and the link http://vincentarelbundock.github.io/Rdatasets/csv/datasets/iris.csv.
2. Using the Sequential interface in Keras, build a model with 2 hidden layers with 16 neurons in each. Compile and fit the model. Assess its performance using accuracy on data that has been train_test_split.
3. Run 2 additional models using different numbers of hidden layers and/or hidden neurons.
4. How does the performance compare between your 3 models?

HW4 Rubric		
Criteria	**Ratings**	**Pts**
Part A		11 pts
Part B		11 pts
		Total Points: 22


Send