MSCI 641: Text analytics

Course information

Grading scheme

- 60% - course project
- 25% - assignments
- 15% - 3 paper reviews

Project (60%)

- Two options:
  - Default Project: Fake News Challenge (FNC-1)
  - Custom Project

- You may work individually or as a group of up to 2 people (encouraged)

Project (cont.)

- Deliverables:
  - Project Proposal. No grade, due: May 30 at 11:59pm.
    - Only for teams doing custom project
    - Teams doing default project: submit team members' names
  - Project Milestone. 5% of the course grade, due: June 27 at 11:59pm.
    - Teams doing custom project: short written report
    - Teams doing default project: train baseline model and submit to the leaderboard
  - Project Final Report. 50% of the course grade, due: July 26 at 11:59pm.
  - Poster session. 5% of the course grade, July 30 (to be confirmed)

Default Project

- Fake News Challenge (FNC-1)
- http://www.fakenewschallenge.org

Fake news, defined by the New York Times as “a made-up story with an intention to deceive”, often for a secondary gain, is arguably one of the most serious challenges facing the news industry today. In a December Pew Research poll, 64% of US adults said that “made-up news” has caused a “great deal of confusion” about the facts of current events

New York Times. “As Fake News Spreads Lies, More Readers Shrug at the Truth”

Default Project (cont.)

- Task: Stance Detection
  - The task of estimating the stance of a body text from a news article relative to a headline.
  - Specifically, the body text may agree, disagree, discuss or be unrelated to the headline

Default Project (cont.)

- **Input**
  - A headline and a body text – either from the same news article or from two different articles.
- **Output**
  - Classify the stance of the body text relative to the claim made in the headline into one of four categories:
    - **Agrees**: The body text agrees with the headline.
    - **Disagrees**: The body text disagrees with the headline.
    - **Discusses**: The body text discusses the same topic as the headline, but does not take a position
    - **Unrelated**: The body text discusses a different topic than the headline

Default Project (cont.)

Example headline

"Robert Plant Ripped up $800M Led Zeppelin Reunion Contract"

Example snippets from body texts and correct classifications

- "… Led Zeppelin's Robert Plant turned down £500 MILLION to reform supergroup. …"
  Correct Classification: Agree

- "… No, Robert Plant did not rip up an $800 million deal to get Led Zeppelin back together. …"
  Correct Classification: Disagree

- "… Robert Plant reportedly tore up an $800 million Led Zeppelin reunion deal. …"
  Correct Classification: Discusses

- "… Richard Branson's Virgin Galactic is set to launch SpaceShipTwo today. …"
  Correct Classification: Unrelated

Default Project (cont.)

Data

Training Set

[HEADLINE, BODY TEXT, LABEL]

Pairs of headline and body text with the appropriate class label for each.

Testing Set

[HEADLINE, BODY TEXT]

Pairs of headline and body text without class labels used to evaluate systems.

Details

Data: The dataset and a brief description of the data is provided at the FNC-1 github.

Source: The data is derived from the Emergent Dataset created by Craig Silverman.

Default Project (cont.)

RULES

RULE #1

For this stage of the challenge, we require all teams to use only the labeled data supplied by FakeNewsChallenge.org (i.e. no external data augmentation is allowed). See also FAQ sections

RULE #2

You may only use the provided training dataset during the development. The test dataset may only be used in the evaluation of your final system. In other words, you may **not** use the test dataset for training or tuning your models.

RULE #3

Have fun!

Default Project (cont.)

- Evaluation metric: two-level scoring system
  - **Level 1**: Classify headline and body text as related or unrelated 25% score weighting
  - **Level 2**: Classify related pairs as agrees, disagrees, or discusses 75% score weighting

Default Project (cont.)

- A simple baseline using hand-coded features and a GradientBoosting classifier is available on Github
- The baseline implementation also includes code for pre-processing text, splitting data carefully to avoid bleeding of articles between training and test, k-fold cross validation, scorer, etc.
- The hand-crafted features include word/ngram overlap features, and indicator features for polarity and refutation
- With these features and a gradient boosting classifier, the baseline achieves a weighted accuracy score of **79.53%** with a 10-fold cross validation.
- This is the baseline you will need to train and submit for your Milestone.

Default Project (cont.)

- The final project will be graded holistically.
- Factors contributing to your grade:
  - creativity
  - complexity
  - technical correctness of your approach
  - thoroughness in exploring and comparing various approaches
  - strength of your results
  - quality of your write-up, evaluation and error analysis.
- Generally, more complicated improvements are worth more.
- Larger teams are expected to do correspondingly larger projects.

Custom Project (cont.)

- Some ideas:
  - Conversational (dialogue) models with machine responses conditioned on a specific persona/character;
  - Text summarization
  - Question answering (SQuAD dataset)
  - If you find an interesting recent paper (can be from the list provided in this course), which does not have a publicly available code, your project may consist of implementing their method and achieving the performance comparable to the results reported in the paper.

Custom Project (cont.)

- Publicly available datasets (corpora) that you can use for custom projects:
  - Amazon product reviews: http://jmcauley.ucsd.edu/data/amazon/
  - OpenSubtitles multilingual movie dialogue corpus: http://opus.nlpl.eu/OpenSubtitles.php
  - Cornell Movie dialogue corpus: http://www.cs.cornell.edu/~cristian/Cornell_Movie-Dialogs_Corpus.html
  - Quora question pairs: https://www.kaggle.com/c/quora-question-pairs
  - Toxic comment classification: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge
  - Yelp reviews: https://www.yelp.com/dataset/challenge

Assignments (25%)

- All assignments will be done using the corpus of Amazon reviews available for download at:

https://github.com/fuzhenxin/textstyletransferdata/tree/master/sentiment

- This dataset contains two classes of consumer product reviews: positive and negative.

- You must use python 3 for your assignments. It is important that you do all assignments, as each subsequent assignment builds upon the work you have done in all previous assignments.

Assignment 0 (0 grade). Due: May 23

- Create a development environment for your subsequent assignments:
- Create a Python 3 virtual environment on your local machine
- Install the following libraries: keras, NumPy, SciPy and gensim

Assignment 1 (code only, 3%) Due date: May 30

- Write a python script to perform the following data preparation activities:
- Tokenize the corpus
- Remove special characters
- Create two versions of your dataset: (1) with stopwords and (2) without stopwords. Stopword lists are available online.
- Randomly split your data into training (80%), validation (10%) and test (10%) sets.

Assignment 2 (code + short report, 7%). Due date: June 6

- Write a python script using SciPy library to perform the following:
- Train Multinomial Naïve Bayes (MNB) classifier to classify the documents in the Amazon corpus into positive and negative classes. Conduct experiments with the following conditions and report classification accuracy in the following table:

Stopwords removed	text features	Accuracy (test set)
yes	unigrams	
yes	bigrams	
yes	unigrams+bigrams	
no	unigrams	
no	bigrams	
no	unigrams+bigrams	

Assignment 2 (code + short report, 7%). Due date: June 6 (cont.)

- For this assignment, you must use your training/validation/test data splits from Assignment 1. Train your models on the **training** set. You may only tune your models on your **validation** set. Once the development is complete, run your classifier on your **test** set.
- Answer the following two questions:
  - Which condition performed better: with or without stopwords? Write a brief paragraph (5-6 sentences) discussing why you think there is a difference in performance.
  - Which condition performed better: unigrams, bigrams or unigrams+bigrams? Briefly (in 5-6 sentences) discuss why you think there is a difference?

Assignment 3 (code + short report, 6%). Due date: June 13

- Write a python script using genism library to train a Word2Vec model on the Amazon corpus.
- Use genism library to get the most similar words to a given word. Find 20 most similar words to “good” and “bad”. Are the words most similar to “good” positive, and words most similar to “bad” negative? Why this **is** or **isn’t** the case? Explain your intuition briefly (in 5-6 sentences).

Assignment 4 (code + short report, 9%). Due date: June 27

- Write a python script using keras to train a fully-connected feed-forward neural network classifier to classify documents in the Amazon corpus into positive and negative classes. Your network must consist of:
- Input layer of the word2vec embeddings you prepared in Assignment 3.
- One hidden layer. For the hidden layer, try the following activation functions: ReLU, sigmoid and tanh.
- Final layer with softmax activation function.
- Use cross-entropy as the loss function.
- Add L2-norm regularization.
- Add dropout. Try a few different dropout rates.

Assignment 4 (code + short report, 9%). Due date: June 27 (cont.)

- For this assignment, you must use your training/validation/test data splits from Assignment 1. Train your models on the **training** set. You may only tune your models on your **validation** set. Once the development is complete, run your classifier on your **test** set.

- Report your classification accuracy results in a table with three different activation functions in the hidden layer (ReLU, sigmoid and tanh). What effect do activation functions have on your results? What effect does addition of L2-norm regularization have on the results? What effect does dropout have on the results? Explain your intuitions briefly (up to 10 sentences).

Paper reviews (3 x 5%)

- Worth 15% of the course grade
- Each student is required to write **three** paper reviews.
- There will be three lists of papers: A, B, C. You will need to choose one paper from each list. The deadlines for reviews are as follows:
  - Review of a paper from List A: July 4
  - Review of a paper from List B: July 11
  - Review of a paper from List C: July 18

Paper reviews (cont.)

- Each paper review may not exceed 300 words. The review should summarize and critique the main contributions of the paper. Specifically, your review should contain the following:
  - Problem. Brief description of the problem or task addressed in the paper. Is this a relevant problem to solve? Is the motivation for this work clear?
  - Related work. Do the authors review all relevant related work? Are any important papers missing?
  - Method. Brief description of the method proposed by the authors. Is the method novel and groundbreaking or an incremental improvement on existing work?
  - Evaluation. What baselines do the authors compare their method to and what metrics do they use? Are the datasets and metrics appropriate? Are the baselines strong or weak? Do the authors compare their method to other state-of-the-art methods?
  - Results. Did the authors improve the state-of-the-art? Is it substantial or marginal improvement? Did they report ablation studies
  - What could be possible extensions or applications of this work?