P2 - Predicting Persuasiveness of Comments

Deadline: Sunday, 20th October (12:00pm)

Phase II: Feature Engineering

In this phase, you are expected to test different features on their power of predicting the persuasiveness of a comment.

You will be provided with 3 datasets for this task:

1. Training dataset: ~5k CMV discussion threads with opinion holders' deltas (label:1) as ground truth.
2. Testing dataset: ~500 discussion threads with ground truth.
3. Crowdsource labelled dataset: ~30 CMV discussion threads from Phase 1 with students' ratings and reasons.

You are expected to use the training set for training your model. And you have to report your results on the testing dataset.

Use the crowdsource dataset to better understand the problem and for coming up with features that could help identify persuasiveness in a threaded discussion. The crowdsource dataset is the one labelled by all students in the class and can give you more perspective about persuasiveness. (look at reasons stated for persuasiveness and ratings for each comment)

Data Preprocessing

As you have seen in Phase I, the discussion texts are noisy – there are system messages from DeltaBot, deleted messages, footnotes from moderators in original posts, non-ASCII characters, etc. To apply NLP techniques such as sentence embeddings and to get solid performance, you need to preprocess the data as the first step.

Please note that in these datasets, the attributes of each comment are extracted from Reddit API, so labels such as CC and RE are not available in the 5k CMV dataset. In this case, you could identify replies from the opinion holders using the author's name attribute. In most cases, there will be only one opinion holder against all CCs. Only in rare cases, there may be someone supporting the opinion holders' view.

There are also more attributes associated with each comment compared to Phase I, such as timestamps of the comments (column unix_time), users' scores (API documentation:

https://www.reddit.com/dev/api/), users' flairs (what's a flair: https://www.reddit.com/r/help/comments/3tbuml/whats_a_flair/).

Requirement: preprocessing the data as you see fit, describe what you did and why you did it in your report.

Feature Extraction

Requirement: Extract the baseline features and propose two additional features related to persuasiveness (explain why you think these features would be helpful in identifying persuasiveness). Vectorize the features for the comments and save them in a file for submission. Include a link to the lexicon in the report for the hedge words and others if any is proposed as additional features.

Baseline features (Include these features for your baseline model):

- the length of a comment
- the similarity between a CC and the OP: use cosine distance of the averaged word embeddings of CC and OP (or some other similarity measure you think might be better suited for this task, if you do just mention it in your report and include some explanation)
- sentiment: use sentiment analysis such as NLTK's sentiment score
- hedge words: find a lexicon online and calculate the occurrence of hedge words in a comment

Some ideas for additional features:

- employ attributes such as user's scores from the dataset
- an additional feature could be a more sophisticated computation for a required feature
- other lexicon-based features such as assertives
- other high-level features such as politeness scores
- Look at the crowdsource labelled dataset to identify some features (may be based on the reasons stated for persuasiveness) or features that could identify persuasiveness in text in general.

Model

The task is similar to P1 from the machine learning perspective. This is a binary classification task in which you classify comments as either persuasive (label:1) or non-persuasive (label:0). You can experiment with simple models like logistic regression or SVM or some more sophisticated models like graphical models or sequential models based on your knowledge. Feel free to even use your earlier models from P1. The focus of this task is on feature engineering and not on machine learning classification. (If you need help on the machine learning classification part, ask the TA.)

Experiment with multiple classification models and choose the best to report results for. You have to report the results for each feature set:

- Feature set 1: baseline features
- Feature set 2: baseline features + two additional features (that you propose)
- Feature set 3: subset (selected by you) of Feature set 2

Evaluation

Following are some evaluation metrics you can use to evaluate the performance of your model with different feature sets during experimentation (You don't need to necessarily include these in your report, look at Deliverables below to see which metrics you have to include)

- Confusion matrix: Analyze false positives and false negatives
- AUC-ROC score as metrics for evaluating model performance since the data is highly skewed
- paired significance test p-value

Requirement: Report the results for the aforementioned metrics and analysis. Specify the libraries if you use any for calculating the numbers.

Deliverables

- Source code
- A report describing the proposed features and model performance. Must include confusion matrix, model accuracy, precision, recall and F1-score, and some other metrics that you deem relevant.
- An output file containing your predicted values for the label. You can just include a csv with a list of your predicted labels in a column (maintain the same order as the test set)
- Readme file detailing all the steps required to run your code. Specify which file contains code for which model, also include all the requirements and libraries needed to run your code.

Deadline for this part is 20th October (12:00pm)