AI in Data Privacy Hackathon

Matthew Dixon

9am October 26th to Noon November 8th, 2018

Deliverables

Please submit a single csv file with the naming convention `<team-neam>.csv` to the google form (which shall be shared with all team members). See sample_submission.csv for an example format for your submission.

You might want to upload the source code for your project to a private repository on GitHub. Winning teams will be asked to provide a link to the repository and present 2 or 3 slides at GCSI 2018 describing their solution approach.

Instructions

1. The dataset is a corpus of privacy notices which have been extracted by a web crawler and a list of scores associated with each document ID. The document's ID appears in the filename.

2. You are first required to extract features from the textual documents, document***.txt . The list of features that you should extract is given in the table below. Further resources and tips for how to extract the feature using python code are also given. We also specify recommended python libraries for performing the task, but you are free to use any approach or tool.

3. Read the description of each feature carefully and integrate the code into your main code so that it generates a feature set which might look like the example feature set in example.csv.

4. Normalize your feature set and combine the labels provided in training_labels.csv. Note that the labels are integers between 1 and 5, representing the strength of the document. 5 is the highest.

5. Split your dataset into training and testing, use cross-validation or otherwise.

6. Apply a machine learning multi-classifier to the labeled, normalized, training set. You might want to start from the multi-logistic classifier example with the IRIS dataset in scikit-learn.

7. Use the F1-score to assess the performance of the multi-classifier.

8. Optionally, assess the model for over-fitting.

9. When you receive the testing corpus (*note* this is *not* provided at the beginning of the hackathon), apply your model to the processed text documents and generate scores. Store your results in a file with document Id and predicted score. See sample_submission.csv. Note that the submission need not be sorted by id.

Name	Description	Additional resources
minor	binary (dummy) variable indicating whether the document includes the word ’minor’	reg. exp. (RE module)
geo-location	binary (dummy) variable indicating whether the document includes the word ’geo-location’	reg. exp (RE module)
contact email address	binary (dummy) variable indicating whether the document includes one or more email addresses	reg. exp (RE module)
Discloses vendors	binary (dummy) variable indicating whether the document includes the word ’vendor’	reg. exp (RE module)
Discloses whether they sell personal data	binary (dummy) variable indicating whether the document describes whether they sell personal data	spaCy. See example of phrase matching in utilities/readability.ipynb
Discloses whether they do not sell personal data	binary (dummy) variable indicating whether the document describes whether they do not sell personal data	spaCy
Discloses how they share the personal data	binary (dummy) variable indicating whether the document describes how they share personal data	spaCy
Discloses whether they do not share the personal data	binary (dummy) variable indicating whether the document describes whether they do not share personal data	spaCy
Uses cookies	binary (dummy) variable indicating whether the document includes the word ’cookies’ or variants	NLTK porter stemmer. See example of stemming in utilities/readability.ipynb
smog index	standard measure of readability	see utilities/readability.ipynb
fog index	standard measure of readability	see utilities/readability.ipynb
avg sentence length	standard measure of readability	see utilities/readability.ipynb
flesch reading ease	see Section 5 of https://arxiv.org/pdf/1809.08396.pdf	see utilities/readability.py
dale chall readability score	see Section 5 of https://arxiv.org/pdf/1809.08396.pdf	see utilities/readability.py

Resources

- Any machine learning book. I like Real-World Machine Learning. https://www.manning.com/books/real-world-machine-learning
- Introduction to NLTK. e.g. https://www.nltk.org/book/ch01.html
- Hackers guide to spaCy: https://nlpforhackers.io/complete-guide-to-spacy/

Scoring

The team with the highest mean f1-score on the submitted model output will win the competition. There are currently prizes for the top three teams who may be required to give a quick presentation at GCSI 2018. Good luck!