Toxic Comments Classification Challenge

Abstract

This project aims to come up with a multi-headed model to distinguish toxic comments on Wikipedia from clean ones, and to identify the types of toxicity present. Currently, we have explored a total of 6 kernels and decided the variable Machine Learning framework and the distinguishing features to be implemented within our model.

Progress (New Section)

Currently, each member of the team has explored at least a kernel each which we have summarised the various distinguishing factors that are implemented in their models, which we will then collate and select the more appropriate factors to implement in our framework.

We did this to learn how other members of the Kaggle community have approached this problem, and the methods that have implemented. From this, we observed that most existing kernels reduced the problem to a binary classification one and used feature engineering as the basis of their framework.

Having done our research, we can discuss the viable methods we have chosen to tackle our main goal – and our various progress regarding them

- Viable ML Frameworks
    - Mainly Logistic Regression, Naive Bayes or some sort of log-Naive Bayes/SVM Hybrid is used to tackle the problem
- Feature Extraction and Engineering
    - Implementation of feature extraction on text. This has proven to be a daunting task due to the difficulties, arising from the amount of irregularities, in converting text to more useful form of data.
    - Explored the use of different variants of feature extraction (bag of words vectorizers, word bigrams) and their respective advantages and disadvantages
    - How to handle data leakages (text information that contains non-textual information like IP addresses, usernames)
    - Modifications to corpus that would increase efficiency
        - homogenising spelling of words, casing, etc
        - removal of unusual characters
        - possibly removal of words with very low frequency
    - Removal of neutral words like articles / pronouns
- Potential Pitfalls and Problems
    - How to handle words that does not appear in training data?
- Validation and Regularisation
    - Pitfalls in conducting validation/regularisation
    - How to segregate data to conduct validation

Motivation

With the evolution of technology, platforms, such as social media, that allows the communication of personal thoughts and feelings are increasingly prevalent. However, this degree of freedom is associated with problems such as promoting hate, hurling abuse anonymously or cyber-bullying - resulting in a toxic online community. Hence, this project aims to study this problem and develop a model that can help identify and classify the various forms of possibly inappropriate comments that stifle productive online discussions.

Statement of the Problem/Task

This proposal aims to develop a multi-headed model which can detect different types of toxicity in comments. These types of toxicity include “toxic”, “severely toxic”, “obscene”, “threat”, “ identity-based hate”.

Possible Approaches: Traditional Supervised Learning techniques (Logistic Regression,Naive Bayes,K Nearest Neighbours), Deep Learning Technologies(Tensorflow,Keras)

Key Question (Main goal):

- Develop and implement an analytical framework in our model to identify the various (could be more than one) types of toxicity within online comments.

Additional Questions (Stretch goals):

- We aim to delve into sentiment analysis to better understand what constitute a toxic comment and how the training data proposes which type of toxicity is present in the comment (i.e. distinguishing between a comment that is a classified as both a threat and toxic, and a comment that is just classified as a threat). We can start with analysis of keywords in each text. This can later be extended to include punctuations, the length of sentences as well as Uppercase letters/words.
- We also intend to analyse how spam comments affect our model.

General Approach

- Formulate the key problem to solve and explore the problem scope.
- Explore possible data analysis tools.
- Engage in basic exploratory data analysis and visualisations. (Univariate ,bivariate data analysis)
- Data preparation and clean-up.
- Feature engineering
- Explore possible classification and predictive models.
- Implement chosen model,
- Review and revise model if necessary.
- Work on stretch goals.
- Explore deep learning technologies if time permits.

Evaluation

Satisfactory Outcome(C-grade):

- Completes project challenge with a working solution.
- Achieve 95% accuracy with our model.
- Addresses key question(main goal)
- Group works through various relevant kernels / tutorials.
    - Traditional Supervised techniques (Logistic regression, Naive Bayes etc.)

Excellent Outcome(A-grade):

- Completes project challenge with an innovative and improved solution(compared to initial proposed solution)
- Multiple experiments with insights on why final approach was chosen.
- Achieve > 97% accuracy with our model.
- Address key question(main goal) + most(if not all) of our stretch goals.

- Explored Deep Learning technologies
- Group works through various relevant kernels / tutorials
  - Traditional Supervised techniques (Logistic regression, Naive Bayes etc.)
  - Deep Learning technologies (Tensorflow, Keras)

Resources

Python(Pandas, Numpy) for exploratory data analysis and manipulation.
Python(Matplotlib , Seaborn) for data visualisation.
Python(SKLearn/Keras/Tensorflow), machine learning libraries to implement models.
Kaggle Kernels(listed in references)

Schedule / Role Assignment

Grayed-out parts are goals that have been met.

Week 5 (10 - 16 Sep)	Define and flesh out scope Complete Project Proposal
Week 6 (17 - 23 Sep)	Explore and analyse dataset (manipulation and cleaning of data) Finalise approach to take Ideation for implementation of the model Complete peer review of project proposals
Recess Week (24 - 30 Sep)	First round of implementation Testing, debugging, improving of first model
Week 7 (1 - 7 Oct) [Midterms]	Exploration of additional questions and other suitable approaches Evaluation of current available model
Week 8 (8- 14 Oct)	Improve on existing model based on what was found out in previous week Interim Report
Week 9 (15-21 Oct)	Implement proposed model + evaluation (Subgroup 1) Explore stretch goals 1|2 (Subgroup 2)
Week 10 (22 - 28 Oct)	Fine-tune / improve proposed model. (Subgroup 1) Explore stretch goals 1|2 (Subgroup 2) **Video, Poster and Report** Report (Subgroup 1) Video + Poster(Subgroup 2)
Week 11 (29 - 4 Nov)	Continue working on Video, Poster and Report. Fine-tune / improve proposed model. (Subgroup 1) Explore stretch goals 1|2 (Subgroup 2) Explore Deep Learning technologies(If time permits).
Week 12 (5 - 11 Nov)	Continue working on Report. Explore Deep Learning technologies(If time permits).
Week 13 (12 - 18 Nov)	Prepare for presentation. Finalise Report.

Due to midterms and conflicting schedules, we have fallen behind quite a bit in our initial goals. However, after the exploration of the kernels, we have a better understanding of the problem we are facing and have made necessary adjustments in our schedule and included task assignments as shown.

Acknowledgements (New Section)

Reviews in general:

- Should consider sentiment analysis.
- More explanation and elaboration on general approach, evaluation and resources. Evaluation part should have a benchmark(e.g. accuracy).
- Key questions not really related to challenge.
- Title should be made more interesting.

We have refined the relevant sections, except for title which we think it's fine as it is.

References (New Section)

[1] Abhishek (2018) Approaching (Almost) Any NLP Problem on Kaggle - [Online]. Available at: https://www.kaggle.com/abhishek/approaching-almost-any-nlp-problem-on-kaggle

[2] Beng (2018) Classifying multi-label comments - [Online]. Available at: https://www.kaggle.com/rhodiumbeng/classifying-multi-label-comments-0-9741-lb

[3] Howard (2018) NB-SVM Strong Linear Baseline - [Online]. Available at: https://www.kaggle.com/jhoward/nb-svm-strong-linear-baseline

[4] Jagan (2018) Stop the S@#$ - Toxic Comments EDA - [Online]. Available at: https://www.kaggle.com/jagangupta/stop-the-s-toxic-comments-eda

[5] Kumar (2018) Logistic Regression TFIDF - [Online]. Available at: https://www.kaggle.com/sudhirnl7/logistic-regression-tfidf

[6] Wang and Manning (No date) Baselines and Bigrams: Simple, Good Sentiment and Topic Classification - [Online]. Available at https://nlp.stanford.edu/pubs/sidaw12_simple_sentiment.pdf

[7] Waseem et al (2018) Understanding Abuse: A Typology of Abusive Language Detection Subtasks - [Online]. Available at http://aclweb.org/anthology/W17-3012