RQ: Can abusive behaviour in OSS be automatically, accurately identified?

We will develop a classifier that can automatically identify toxic comments.

Step 1: Develop initial training set.

- Positive class (toxic): a manually coded list of insulting sentences posted in the Linux Kernel Mailing List (LKML) created in a previous study [1]. Avaiable at: http://flossdata.syr.edu/data/insults/
- Negative class (non-toxic): The most polite comments found using a state-of-the-art natural language processing tool that automatically classifies text by its politeness [2]. OSS comments will be obtained through the GitHub API. GitHub is the most popular software development platform and its API will allow us to collect online communications from a range of OSS projects. Alternatively, GHTorrent can be used to obtain OSS comments [3]. If using GHTorrent, we must use the mongo DB as the mysql DB truncates the comment text. To start, let's collect data for https://github.com/nodejs/node to get started (more later).

Step 2: Create classifiers:

We will trial a support vector machine or a random forest classifier. Xuyun will guide you on classifier selection and feature engineering.

Step 3: Active learning

Since the dataset of insulting comments covers only one project, we will use active learning to uncover additional text that will need to be annotated by a human to create a more comprehensive training set. To employ active learning, a classifier is developed, and additional data is annotated by selecting the unlabelled data that obtains the least confident classification [4]. This ensures that the comments that we manually annotate will differ from each other, enabling us to find a wide range of insults and abusive comments. We will continue employing active learning to increase the training set until theoretical saturation is reached and no new types of insults or abusive comments are being uncovered. Figure 1 illustrates our approach to do this. We will need additional OSS comments for more projects before running the active learning.

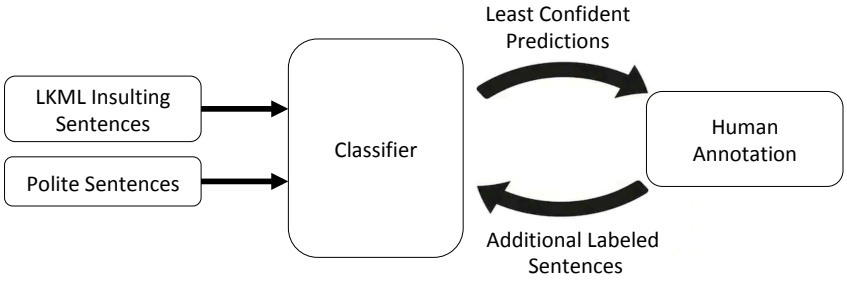

Figure 1. Active Learning to identify comprehensive set of abusive, insulting sentences.

References

[1] Squire, M. & Gazda, R. (2015). FLOSS as a Source for Profanity and Insults: Collecting the Data. In *Proc. of HICSS-48*. IEEE. Hawaii, USA. 5290-5298.

[2] Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J., & Potts, C. (2013). A Computational Approach to Politeness with Application to Social Factors. *arXiv preprint arXiv:1306.6078*.

[3] GHTorrent: http://ghtorrent.org/

[4] Dhinakaran, V. T., Pulle, R., Ajmeri, N., & Murukannaiah, P. K. (2018, August). App Review Analysis Via Active Learning: Reducing Supervision Effort without Compromising Classification Accuracy. In *Int'l Requirements Engineering Conference* (pp. 170-181). IEEE.