Proposal: To Create a Chatbot for Detecting Hate Speech

Wesley Aldridge
waldridge@knights.ucf.edu

Miles C. Crowe
miles.crowe@knights.ucf.edu

Mauricio De Abreu
mabreu@knights.ucf.edu

1. MOTIVATION & PROBLEM STATEMENT

Hate speech is defined broadly as aggressive language targeted at a person for attributes typically beyond their control. Traits such as nationality, gender, race, sexual preference or disability are popular targets of hate speech[1]. Aggression towards the victim may manifest as insults, personal attacks or even threats. As online communication between people is becoming more commonplace, exposure to hate speech is almost certain to reach more people. Unfortunate as it is, this allows the spread of underlying beliefs that contribute to such behavior.

The motivation for this project is rooted in the desire to curtail the spread of hateful language. There are countless paths for hate speech to spread. Providing a human intervention is infeasible given the amount of communication that occurs over the internet. Naturally, this presents an excellent opportunity to contribute an automated approach to detecting and flagging hate speech by utilizing NLP in real time.

2. RELATED WORK

Our work will apply a Twitter hate speech dataset from Manoel Horta Ribeiro, et al., [2][3]. They used this dataset to characterize Twitter users as either hateful or not, in order to analyze patterns and trends among the users labeled as hateful. Instead we will use this dataset to create a Discord realtime hate speech detection bot.

At this stage, you don't want to provide a comprehensive list of related work, but rather you want to consider this as a part in which you will provide a list of the resources that will be used to assist you conducting the project. Find out some of the related work that would be relevant to this project, and summarize how similar or different your work to them is. Even better, highlight in broad terms what would the δ you think you will achieve by this magnificent work be.

Challenge 1

https://homes.cs.washington.edu/ msap/pdfs/sap2019risk.pdf

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

3. STOPPING THE SPREAD

This project aims to curtail the spread of such speech by detecting hateful dialog and providing an alert mechanism to raise events for handling the occurrence automatically as it happens. Clearly stated, the chat bot would be a silent participant in a chat, regardless of the number of participants, which would monitor the text exchanges. When hate speech is detected, the event would be passed to a host framework or external agent to handle the event.

A simple implementation of this would be a chat bot, connected to a Discord server which is trained utilizing a robust hate speech data set from these sources: [4][5]. A neural network would be trained to detect hate speech. This neural network would be the critical part of the chat bot which would read user messages and detect hate speech in real time, and alert the moderators and admins of the Discord server.

4. EXPECTED OUTCOMES AND RISK MANAGEMENT

As in most software projects, expectations will need to be curtailed and priorities need to be established. At the bare minimum, this project should yield a simple input/output text classifier that is accurate with meeting the expectation of identifying hate speech. Formally stated, this implies that core functionality should be established prior to moving towards the novel aspects such as portability and integration with external platforms. This can be realized though a simple design that exposes only the necessary API to validate and test the outcome of the core goal. Once this goal is achieved, a narrow set of integration targets can be established for demonstration purposes.

In terms of the main goal, there is risk of biases and over fitting of the training data. Over-fitting errors will need to be carefully analyzed and bias will need to be mitigated with tools we will discover later in this course.

Another risk would be training data mismatch with regards to the platform. For example, users in Discord are allowed 2000 characters vs 280 characters for a Tweet on Twitter. Size constraints may dramatically affect how hate speech is formulated due to the compression of ideas on platforms that severely restrict text lengths. Careful validation can easily address this by restricting the test data to mimic similar sizes to that of the training data.

Lastly, temporal changes in hate speech may present classification errors as slang and dialect may change over time. This risk will have to be accepted unless fresh and up to date training data becomes immediately available.

5. PLAN AND ROLES OF COLLABORATORS

Wesley will code the neural network, train it with the hate speech data, and test it. The neural network will be what the bot uses to label messages as hate speech or not hate speech.

Miles will integrate the neural network into an online hate speech detection service that he will code and test. The service will be used by the Discord bot to do its message analysis and hate speech detection.

Mauricio will incorporate the online hate speech detection service into a Discord server bot that he will code and test. The bot will detect hate speech in Discord server messages in real time.

Everyone will work on the write-up and presentation.

Divide your project into components, and tell me who is going to work on what, how much time each will work on each item. Have a timeline for the project. Tasks may include coding, testing, evaluation and analysis, write-up, presentation, etc.

6. REFERENCES

[1] "Dictionary.com." https://www.dictionary.com/browse/hate-speech. Accessed: 2020-01-21.

[2] "Hateful users on twitter." https://github.com/manoelhortaribeiro/HatefulUsersTwitter. Accessed: 2020-01-23.

[3] M. H. Ribeiro, P. H. Calais, Y. A. Santos, V. A. Almeida, and W. Meira Jr, "'like sheep among wolves': Characterizing hateful users on twitter," *arXiv preprint arXiv:1801.00317*, 2017.

[4] "Hate speech." https://www.kaggle.com/mohit28rawat/hate-speech#labels.csv. Accessed: 2020-01-22.

[5] "Hate speech." http://hatespeechdata.com/. Accessed: 2020-01-22.