Senior Design 4739 05/06/2020
Professor Ashis

CyberBullying

Detection System

Giovanni Berrios

Chanhee Shin

Nishal Kallupalle

Team Information:

Team Name/Project Title: Cyberbullying Detection and Transformation System

Department: Computer Science

Faculty Advisor(s): Ashis Biswas

Primary Team Contact, email: Chanhee Shin, chanheeshin@live.com;

Team Members (first name, last name, and e-mail): Chanhee Shin, chanheeshin@live.com;

Giovanni Berrios, giovanni.berriosfigueroa@ucdenver.edu; Nishal Kallupalle,

nishal.kallupalle@ucdenver.edu;

Introduction:

This semester was mainly predicated on identifying the problem we would be trying to solve. We ended up deciding on detecting bullying through online messages also known as cyberbullying. We drew inspiration mainly from Team Advisor Ashis Kumer Biswas who proposed the basic idea of detecting bullying in messages to us. The team went ahead and rolled with the idea and evolved it further. The team decided on a final goal of creating an application that would be linked with various social media sites, for now focusing on Facebook. For the app to be linked with the social media sites, users would have to allow permission for the app to read their messages and possibly make changes to the messages they send as well. For example, if the app were to detect a high probability bullying message, than the app would itself alter the message so it would be perceived by the reader as a positive message.

Background:

During the first section of this semester, a lot of background research was done on cyberbullying. The first question we may ask is What is Cyberbullying? Well, Cyberbullying is is the use of technology to harass, embarrass, or threaten another entity. Cyberbullying can be either intentional or intentional. Although ¼ of teens are bullied on average, only ⅛ of them admit based on the fear of being punished. The effects of Cyberbullying include mental disorders such as depression, suicide, Anxiety, and stress related disorders. Signs of Cyberbullying can be seen if the person at hand is emotionally upset after using phone/internet. They are extra secretive/protective of digital life. They are experiencing withdrawal or changes in mood/behavior. The person may be jumpy or nervous when receiving messages. Measures to protect against cyberbullying include blocking said bully, limiting access to technology, and knowing what is normal. Preventative measures that can be taken to prevent Cyberbullying including talking to children about what Cyberbullying is, havng children practice real-world social skills, being able to openly communicate with children, and understand what devices children are using.

Cyberbullying examples include:

- Sending Mean emails,texts, and/or instant messages.
- Neutral messages to someone on an extreme basis to the point of harassment.
- Posting hurtful things about someone on social media.
- Spreading rumors.

Some youths who are at an increased risk of Cyberbullying are those with learning disabilities and those with physical disabilities. Unique properties of Cyberbullying include that it can be done anonymously, may have large audiences, easier since consequences aren't visible, and harder to manage by adults.

Types of Cyberbullying are:

Outing- Posing private/sensitive information to embarrass others.

Fraping- Getting access to victims social media and impersonating them to be funny or ruin reputation.

Dissing- Sharing/posting cruel information about someone to ruin their reputation/friendships with others.

Trolling- Provocation for response.

Trickery- Two parts. First is gaining trust so secrets can be revealed which is then posted. Or the second in which pretending to be a close friend to expose them.

Sockpuppets/Catfishing- Gaining trust through fake account, then sharing with others so that the "others" can bully them (sockpuppeting). Catfishing is the same but romantically.

Doxing- Sharing private information such as SSN, Credit cards, phone numbers, other personal stuff.

Encouraging self harm- This is when someone encourages another to commit harm to oneself.

Project Description:

A software design and implementation, to allow harmful messages that were sent by a user to be detected by the system. The system will score each message it receives and decide if it is harmful or not. If deemed harmful, the system will then proceed to use a machine learning algorithm in order to change the contents of the message into a “positive” message as deemed by the system. The transformed message is then received by the end user with little to no trace of the original harmful content.

Project Abstract:

Cyberbullying Detection implements our coded, machine learning algorithms, in finding a negative comment from the messages it receives by a user. The algorithm first gives the message a value and then based on our pre trained data, it decides if the comment is harsh enough to be transformed or not. If it is indeed harsh, then the system will look through our complex network of users and find how this user talks to people on average and how they talk to the end user on average. Based on this data, the system will decide if the message needs to be transformed. If so, the message is run through a series of models in order to change negative components of the sentence into positive components. The transformed sentence is then checked by our initial algorithms. It is assigned a value and if the value results in a positive sentence, the system will proceed to send the transformed positive sentence to the end user. Otherwise, the sentence will

be placed through the models again. The users communicate through a developed web front face and they are connected to a central server. The users are termed as clients. If any messages are modified the receiving user will be notified along with the modified message.

Front - End Explanation:

The front end operates off of the python tKinter module to build the primary skeleton of the ui. Users start at a simple login screen where they input their credentials and then press the login button which then automatically guides them to the primary chatting interface. From there the application has a simple display box, a text box to enter messages, and a button to send the messages.

Back - End Explanation:

The back end works utilizing two different machine learning neural networks. The first network is a feed forward sentiment analyzer that determines whether a given message is positive or negative. This is done by using the top 500 words within our dataset in order to create a bag of words with a tf/idf normalization. This is then fed into our four layer neural network which then uses the frequencies of each word to determine whether the sentence is positive or negative. After a message has been determined to have a negative sentiment, it is then fed into a second neural network that is responsible for transforming the message into a more positive message. Before the message can be fed into the second network however, it must first be vectorized into a one-hot-encoded character array. This second network then utilizes a recursive neural

network with a long-short-term memory in order to predict what the best output would be given the current state of memory and the message given. After completing the transformation, the message is restructured back into a string of letters based on the output sequence given and the final message is sent to the recipient.

Results:

After conducting various tests on network structure and data refinement, our results were both exciting yet also sub par. Our first network, the sentiment analysis network, worked well in identifying whether a message was positive or negative, but still has room for improvement. This was still great though as it allowed us to move onto our second phase of transforming messages that were deemed to be negative. The results obtained from this phase were both good and bad. The dataset used required immense amounts of cleaning which proved to be difficult for various reasons. Grammar was a major issue as well as chat lingo as it made the data very difficult to clean up in its entirety. Despite this, we were however able to make a model that showed strong promise as many messages that were negative were transformed into more positive messages, however, sometimes the results were a not fully coherent message. There is plenty of room for improvement but what was built is a solid foundation that can be improved upon to hopefully make a fully functional product.

Conclusion:

All in all, this past year was a great learning experience. The team improved it’s knowledge vastly especially in terms of cyberbullying and understanding Machine Learning and it’s

algorithms. The Team had a lot of fun researching and implementing what we researched. We had a great time reading comments from reddit. Roastme was a great time. Can’t wait to continue our endeavor improving our system and making it more accurate in the future.