An Analysis of Cyberbullying Detection with Machine Learning Methods

Justin M Johnson, Z231236514

CAP6673 - Data Mining and Machine Learning

Florida Atlantic University, Spring 2018

Table of Contents

Introduction

Cyberbullying, a form of bullying which occurs through electronic communication, is a global issue that is increasingly more difficult to monitor as social media platforms continue to grow. Unlike traditional bullying that would occur during specific hours (i.e. school hours), cyberbullying occurs 24/7 and imposes new challenges as users make anonymous or fake profiles to conceal their identity while attacking their victims. In 2017 Twitter alone maintained over 300 million active users [9], making it extremely difficult for human detection systems to efficiently identify cyberbullying at scale. Data mining and machine learning systems are capable of detecting and extracting knowledge from large amounts data, proposing as a solution to the automated detection of cyberbullying.

The proceeding report documents the current state of cyberbullying detection via machine learning methods. A total of six case studies were selected between the years of 2011 and 2018. Some of the major challenges associated with cyberbullying detection through machine learning are introduced. Then the methods employed by each research team are examined and presented in a cross-sectional manner. Rather than presenting one research paper in full at a time, the studies are divided up into their core components: data collection and labelling, pre-processing and feature selection, then learning methods and their results. As each area is a problem in itself, presenting the studies in this manner will allow the reader to better compare the multiple solutions associated with each challenge. Throughout the analysis, proposals will be introduced for future work.

The Cyberbullying Problem

Cyberbullying and its current impact on our society is presented to serve as motivation for further research. Love Engman stated in 2015 that “54% of young people on Facebook had experienced cyberbullying.” [3] In fact, all research groups included evidence of cyberbullying affecting more than half the teenage population worldwide. Victims of bullying often suffer from both mental and physical side effects, with symptoms ranging widely from mild depression to isolation to as far as self-harm and suicide. Some have even deemed cyberbullying as a “national epidemic” [10]. Software systems that can efficiently and accurately detect cyberbullying throughout social media, forums, and chatting services will have a significant positive impact on future generations, potentially increasing the overall well-being of many adolescents and young adults throughout the world by minimizing victimization.

Challenges of Cyberbullying Detection

There are many challenges in the automated detection of cyberbullying. Many of these problems will stem from the fact that this is primarily a natural language processing problem, a field of study with many inherit challenges. Generating features that capture the essence of bullying, or the meaning of any language in general, is an extremely difficult problem. In most (non-toy) machine learning problems, the acquisition of large amounts of quality data is another major challenge to overcome. With the expansion of the web and IoT, more and more data has become available, but there is still a significant cost in manually collecting this data. Another challenge that will be common among all research groups is a strong data imbalance, as only a small fraction of the data will contain bullying.

Natural language processing (NLP) is the field of computer science and artificial intelligence concerned with processing and understanding the human language. The complexity of human language and its many ambiguities makes it extremely difficult to model. There are thousands of languages in existence, not to mention the many dialects and regional differences. Word meaning is usually context specific, words representative of bullying in one context may take on completely new meaning in another context. There are also cultural influences on language, where one group of individuals may consider a conversation as acceptable and a different group would deem the language as inappropriate or offensive. Digital communication often involves an entirely new vocabulary consisting of slang and abbreviations, with new words being introduced on a monthly basis. A concept like sarcasm may be easy for a human audience to detect, but it creates many challenges when presented to a computer. These are just a few of the problems revolving around natural language and digital communication, all of which apply to cyberbullying detection.

The largest task in any supervised machine learning problem is the collection and cleaning of data, often occupying up to 80% of the project. In this particular study of detection of cyberbullying, there is no large, publicly available data set prepared for training machine learning models. With a lack of data, the research groups are forced to query and crawl social media platforms to gather their data and then employ human labelers to annotate the data with appropriate labels. This method works, as will be presented in the data collection and labelling section, but it is very expensive, and their training and test data quality and size will be constrained by both the team's time and resources. Most of the great success in recent machine learning problems, like speech or image recognition, was achieved through the availability of massive data sets and increased computational power. The data set quality and size will always influence the performance of the prediction model.

Introduction to Case Studies

Six case studies with a focus on cyberbullying detection through machine learning were selected for review. The studies span between the years of 2011 and 2018, providing a variety of solutions to the previously mentioned challenges in cyberbullying detection. The following table introduces the papers along with their date of publication, authorship, and a short description:

Group	Title	Date Published	Authors	Description
A	Modeling the Detection of Textual Cyberbullying	July 2011	Dinakar et al.	Apply binary and multi-class classifiers on a corpus of 4500 YouTube comments.
B	Using Machine Learning to Detect Cyberbullying	December 2011	Reynolds, K.	Use both rule-based and bag of words models on labelled Formspring.me data.
C	Automatic Detection of Cyberbullying on Social Media	June 2016	Engman, L.	Performs two experiments on Ask.fm Q&A dataset and compares their results.
D	Mean Birds - Detecting Agression and Bullying on Twitter	May 2017	Chatzakou, D. Kourtellis, N. Blackburn, J. De Cristofaro, E. Stringhini, G. Vakali, A.	Extract text, user, network, and session-based features from a Twitter corpus, then apply various classifiers to the problem of detecting cyberbullying.
E	Cyberbullying Detection with Weakly Supervised Machine Learning	July 2017	Raisi, E. Huang, B.	The “first specialized algorithm for cyberbullying detection that allows weak supervision and uses social structure” to detect user rolls.
F	Detecting Cyberbullying and Aggression in Social Commentary Using NLP and ML	January 2018	Sahay, K. Singh Khaira, H. Kukreja, P. Shukla, N.	Performing binary and multi-class classification, using YouTube comments and Twitter messages for a combined training data

When comparing case studies and their strategies in the proceeding sections, they will be referred to by their group identifier (A - F). The full reference for each case study can be found in the references section. Note that the groups have been ordered by date of publication.

Data Collection and Labelling

Given sufficient quality data, machine learning models can be trained to make predictions on future data with moderate to good accuracy. One of the major challenges in the automated detection of cyberbullying is the acquisition of quality data that is representative of cyberbullying. In all of the case studies reviewed, training data had to be manually collected, labelled, and cleaned. Group E was the only group to re-use prior research data, and they used it in addition to data manually collected and labelled.

Group A collected data by scraping the comments off of YouTube pages which contain controversial videos, as the controversial videos are more likely to contain aggressive comments. A total of 50,000 comments were collected, and then grouped into clusters: physical appearance, sexuality, race & culture, and intelligence. 1500 comments were then selected from each cluster and manually annotated by two individuals, creating the final data set.

Group B collected their data from a question and answering forum with higher than usual percentages of bullying, Formspring.me. 18,554 user pages and their posts were collected by crawling the site, along with user profile details, outbound links, and page meta data. Workers were employed through Amazon's Mechanical Turk service to label the data. The workers labelled the data by answering questions which categorized the data as bullying vs non-bullying, listed the words indicative of bullying, and provided a 1 to 10 ranking of bullying severity. Group B's final data set contained 13,652 posts, 792 of which (5.8%) contained cyberbullying.

Group C collected their data from ASK.fm, another question and answering website where users ask questions on other users' pages. Approximately 24,000 questions and answers were gathered by recursively crawling the site. The researcher then manually labelled all

instances as bullying or non-bullying. Only a small percentage of the data (278 comments) were labelled as bullying.

Group D used the Twitter API to collect 1.65 million tweets between June and August of 2016. The first million tweets are random tweets to serve as the baseline. The following 650K tweets were collected using 309 hashtags that correlate to bullying/hate speech vocabulary. The 309 hashtags correlated with bullying were generated by sampling the baseline tweets for tweets containing hashtag #GamerGate and extracting other hashtags found on these tweets. #GamerGate is a very controversial movement in the gaming industry that is known to contain large amounts of offensive speech/bullying.

Group E collected data from Twitter, ASK.fm, and Instagram. For Twitter, tweets were collected between November and December of 2015 by querying the Twitter API for curse words listed in a lexicon. In addition to the tweets, all conversation threads connected with the tweet were collected. A baseline of random tweets was then collected using a crawling pattern, resulting in a final Twitter set of 296,308 tweets across 180,355 users. 2,863,801 question and answers were collected from ASK.fm. The ASK.fm data set was reused from Hosseinmardi et al. [7], except that anonymous users were removed because the group's methodology was not equipped to handle anonymous users. Instagram data was also borrowed from Hosseinmardi et al., collecting for each user: media shared, users who've commented on media, and the comments on the media. This group also used Amazon's Mechanical Turk to label their data, but in a slightly different manner. Unlike previous groups which have classified their comments as bullying or non-bullying, Group D has conversations between two individuals labelled as bullying or non-bullying. They also have their top scoring keywords labelled by Mechanical Turk as indicative of harassment or not.

Group F collects 3,947 comments from YouTube and 2647 tweets from Twitter. They also utilize a data set available on Kaggle by Imperium as their test data set. This data set was made available for a machine learning contest in 2010. All data is labelled manually by the team as “bully”, “non-bully”, “spam”, and “aggressor” for multi-class classification, and as “bully” and “non-bully” for binary classification. Group F has the smallest data set in comparison to other groups. Unlike Group D who removed spam from the data set, this group labelled data as spam and kept it in the data set.

There are some common trends between all research groups. For the most part, all data is collected through custom software designed to crawl social media platforms and forums. Some groups labelled the data themselves, and others outsourced to a service for labelling (specifically Amazon’s Mechanical Turk). One group was able to obtain data from a previous research team, and another group was able to locate data from a previous Kaggle competition. Several groups used seeds, or queries, to point their crawling software in the right direction, collecting data specifically relate to queries around controversial topics or offensive words. Only two of the six groups combined data from multiple platforms. I believe that each platform’s communication protocol is different, and better training data could be generated by utilizing more platforms simultaneously. Many groups collected data over a few months, which can introduce a bias as trends and changes in language come and go, passing through social media platforms over time. I believe that an existing cyberbullying detection model could be utilized in a semi-autonomous continuous data collection process. The data collected would be easier to manually label as the model would have filtered out many non-bullying examples. Running a system for significantly longer, a year perhaps, would generate significantly more data, data that was more representative of the internet’s broad spectrum of bullying. More data would allow researchers to utilize more

complex deep learning models, models that have achieved extremely high accuracy in tasks such as speech recognition.

Pre-Processing & Feature Selection

Data features are the characteristics of the individual data samples, they describe the data and they are used by the machine learning models to make predictions. Having collected a set of cyberbullying data, it must now be engineered to produce an effective feature set that can be used as input to the machine learning models. How can language complexities best be represented, such that a machine can effectively draw meaning from it? Cyberbullying is a form of digital communication, so one obvious feature set is the textual context, often represented with a bag of words model that considers weighted term frequencies. Text should be cleaned and regularized to obtain best results, methods which will be included to follow. Additional features that can help in describing cyberbullying include: user usage statistics, followers, media content, social network (graph) features, and more. These feature types and others will be covered as the group's research is presented.

Recall that Group A's focus was to classify YouTube comments into three sub-categories of bullying (sexuality, race/culture, and intelligence) using binary and multiclass classification models. Data was cleaned by removing stop words, stemming, and removal of both repetitive or special characters. Common features that are shared between all three classes of bullying and features unique to each specific class were created. The shared features consist of TFIDF weighted unigrams, a lexicon (Ortony) of words indicative of negative connotation, a list of profane words, and frequently occurring POS bigram tags that were observed in the training set. Part of speech tags will allow the model to detect personal pronouns and other parts of speech common to bullying scenarios. Label specific features consisted of label specific unigrams and bigrams that had high frequency in the classes training set.

Group B curated a list of bad words (296) from the website www.noswearing.com and then manually labelled each word with a severity value from the set {100, 200, 300, 400, 500}, where a value of 500 was given to words of the highest severity. For each data sample, the following features were identified:

1. Number of curse words at each level (NUM100, NUM200, … , NUM500)
2. Percentage of curse words at each level (NORM100, NORM200, … , NORM500)
3. Total number of words in the post
4. The weighted sum (SUM) of the curse words
5. Boolean value that is set true if user is posting anonymously

Group B's feature set is not a traditional bag of words model like Group A had used, but it is similar in the sense that term frequencies are weighted, and word position is assumed independent. The feature set is much smaller than traditional bag of words, because only 296 terms are used for the vocabularly, instead of the corpus' full set. I would suggest that this feature set is weaker than Group A's, as any context has been removed and there is no sense of directed communication via part of speech tags. One advantage this does have over TFIDF weighting is that some words are flagged as more severe than others, as assigned by the research team. Introducing domain knowledge to feature extraction should almost always improve results.

Group C collected a very unbalanced dataset from Ask.fm, with a total of 24K comments, only 278 of which were manually labelled as bullying. Data was pre-processed by removing special characters, replacing links and usernames with keywords, and case folding. The first set of features for Experiment 1 was created using the Stanford Parser and 3 dictionaries of strong and weakly offensive words. The Stanford Parser is able to identify grammatical dependencies between words. This is used in combination with the dictionaries of offensive words to identify

dependencies between offensive words and users, providing a feature that denotes offensive speech in relation to a person. Experiment 2 consisted of creating 4 feature sets (A - D), then making attempts at classification with an SVM to compare feature sets. Feature set A is a bag of words TFIDF feature set. Feature set B transformed set A by multiplying the offensive words by a scalar, increasing the weight of offensive words. Their method for expanding the offensive word set was interesting, as they used a seed set to identify additional words using a nearest neighbor approach with word vectors. New words in vector space were labelled as offensive if 3 of 10 of its closest neighbors were included in the seed vocab. Feature set C introduces four new features: the total number of bad words, the total number of second person pronouns, and the total number of third person pronouns, and the total number of positive words. This approach is better than only including offensive words, as comments can include a combination of good and bad words whose structure change the overall sentiment. The fourth and final feature set D uses the Stanford Parser to generate word relation features. A total of 80 word-relation features are generated with the hope of combatting traditional TFIDF by including features that map word dependencies.

Group D experimented with a large set of Twitter tweets. Stop words, spam, URLs, and punctuation were removed or replaced with keyword identifiers during pre-processing. As will be reviewed in results, removal of spam was effective in improving model performance. A combination of user, text, and network-based features are extracted. Some examples of user-based features include total number of tweets, if account is verified, the number of lists subscribed to, and whether or not the user uses default profile image. Features are analyzed, and it is noted that normal users subscribe to more lists than aggressive, bully, or spam accounts. Session based features including total number of sessions, median session time, and time's

standard deviation. These features are compared, and it is noted that there is no significant difference between user types. Text features include more than just words used: number of hashtags, number of emoticons, uppercase text, and number of URLs. It was noted that normal users tend to post less URLs and less hash tags. Semantic and syntactic features are extracted through the use of word vectors, as word vectors capture more than just the meaning of a word. Words of similar part of speech, similar sentiment, used in similar context, etc. will have word vectors with close similarity in vector space. Group D is the first group studied that used word vectors to capture semantic and syntactic meaning of input data. Word embeddings are not new, but they are growing in popularity in the NLP research area as the power of neural networks increases, as neural networks are able to generate word meaningful vector representations given large amounts of input data. Next a sentiment analysis tool, SentiStrength, is used to extract sentiment values (pos / neg) for each tweet. A lexicon of hate speech is used to identify hate speech in tweets, creating additional features. Unfortunately, these features prove relatively ineffective because the short text nature of tweets encourages internet slang that is uncommon in standard lexicons. A variety of social network-based features are now extracted: popularity, reciprocity, power difference, centrality scores, hubs and authorities, influence, and communities. After fully analyzing the effectiveness of all features considered, the most statistically significant features were selected, creating a final feature set of 30 features. Group D has explored more features than any group previously introduced, and the analysis of features has allowed them to maximize feature space while excluding insignificant features. This is a most efficient, as insignificant features will slow down the training process and potentially introduce noise. The feature set is critical, and it is clear that Group D invested a great deal of energy into generating effective features.

Group E has a unique challenge that other groups did not face, and this is the fact that they've combined data from 3 different social media platforms. Another major difference for Group E is that they're employing a weakly supervised methodology where they provide a seed set of word indicative of bullying, then use these seeds to detect bully users and new words indicative of bullying. N-gram text features are the primary feature used in their PVC model. A message between two users is given a weighted score that is calculated by summing bullying-related word scores. The PVC model predicts both a bully score and a victim score using these weighted term features.

Group F used a combination of Twitter and YouTube data for their experiments. Data was cleaned of special characters, case folded, regularized with form reduction, normalized, and grammatical conflicts are resolved. A dictionary of bad words from http://urbanoalvarez.es is used to identify offensive words. Both TFIDF vectors and count vectors are generated for n-grams in the range [1, 5]. Then SelectKBest feature selection, which uses Chi-Squared test to select top K scoring features, is used to generate a final fit data set of shape (6594, 4600), where 4600 is the total number of features. TFIDF is known to give better results than count vectors but considering the ease at which count vector features can be generated it makes sense to create the features and let a feature selection algorithm decide their significance. Group F did not present any unique feature engineering strategies, but relied on TFIDF and count vectors, two popular NLP methods.

In review, a variety of feature selection methods were utilized. The bag of words model with TFIDF weighting was seen several times, and this is no surprise as it remains a very popular NLP strategy. It does suffer from the major assumption that words and their position are independent, as this is clearly not true, words in text have short and long-term dependencies on

each other. Word vectors were introduced by both Group C and Group D, both of which used the technology in a unique manner. Group C used word vectors to identify new words indicative of cyberbullying, using a nearest neighbor-like approach. Group D used word vectors instead of a bag of words model with weighted term frequencies, because word vectors contain much more meaning about a given word. Word vectors are able to encapsulate syntactic, semantic, part-of-speech, and context attributes about a word. Word vectors are generated themselves through machine learning algorithms, typically through neural networks, but there are pre-computed word vectors available for public use online [8]. Group D's approach was very thorough, as they examined many potential features, thinking passed the raw text and offensive words, considering user and user session patterns. The only problem with their feature selection method, is that it restricts them to Twitter data, as they consider many features that are unique to Twitter cyberbullying. This approach is great if attempting to train a model that specifically detects cyberbullying on Twitter, but this model will be useless on other platforms where the same features are not available.

Machine Learning Algorithms and Results

A variety of supervised and unsupervised machine learning methods were used by the 6 groups presented. In general, most groups focus on the problem of binary supervised classification, predicting whether a given text contains cyberbullying or does not. Different learners provide different advantages, and so most groups use the approach of experimenting with a variety of learners and then evaluating their results. As previously mentioned, the biggest challenge is often in the data collection, cleaning, and labelling. Once the data is ready for training, a number of machine learning algorithms can be experimented with relatively easily or learning algorithms can be combined in ensemble methods to improve performance. Learning methods and their results will be compared but note that comparing one group's results to another's does not necessarily provide insight into overall performance, as each group is using data sets of differing quality and different methods of feature extraction.

Group A performed both binary and multi-class classification in order to compare each's efficacy. The group experimented with 4 different learners: Naïve Bayes, Rule-Based JRip, J48 Decision Tree, and a Support Vector Machine. The group evaluates their models using both accuracy and the kappa statistic, as the kappa statistic takes into consideration agreement by change. Group A divided their data set into 50% training, 30% validation, and 20% test. The

below table outlines their results:

	Naïve Bayes		Rule-based Jrip		Tree-based J48		SMO (SVM)	
	Accuracy	Kappa	Accuracy	Kappa	Accuracy	Kappa	Accuracy	Kappa
Sexuality	66%	0.657	**80.20%**	0.598	63.40%	0.573	66.70%	**0.79**
Race	66%	0.789	**68.30%**	0.789	63.50%	0.657	66.70%	**0.718**
Intelligence	72%	0.467	**70.39%**	0.512	70%	0.568	72%	**0.7723**
Mixture	63%	0.445	63%	0.507	61%	0.456	66.70%	0.653

Note that the binary classification methods were more accurate than the multiclass classifier. The Rule-based Jrip performed the best in regard to accuracy, and the SVM performed best in regard to the Kappa statistic. In general, accuracy is a poor method of measurement, especially when the data set is largely imbalanced. Precision, recall, and AUC should be considered when evaluating models.

Group B used their Formspring.me data sets with a combination of learners to compare classification results for both rule-based models and bag of words models. The group noted that their training data contained just 6% bullying labels, increasing likelihood of false negatives. They combatted this by oversampling the positive instances (bullying). The group's rule-based experiment scored significantly higher than the bag of words experiments. The group was also able to conclude that their NORM data set outperforms the NUM data set. Recall that the NUM features considered offensive word frequency, and NORM considered offensive word percentages. It makes sense that NORM would perform better, as the length of the input document and the quantity of non-bullying words should be taken into consideration. The

following figures presents their results for the NORM feature sets, both including (top) and excluding (bottom) anonymity features.

	NORM Data Set - WITH ANON									
		Times Positive Results Repeated									
		1	2	3	4	5	6	7	8	9	10
J48	% correctly labeled positive	29.20	47.50	53.30	59.10	63.10	65.60	66.60	67.20	67.40	67.40
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	23.40	36.53	37.71	39.33	39.56	38.62	36.48	34.19	31.74	29.29
	# of Leaves	41	101	117	135	154	172	190	202	206	208
	size of tree	77	192	222	256	293	327	362	384	389	395
JRIP	% correctly labeled positive	29.90	43.40	53.30	59.10	62.10	62.50	63.70	64.00	65.40	65.30
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	24.10	32.43	37.71	39.33	38.56	35.52	33.58	30.99	29.74	27.19
	# of Rules	7	10	18	18	16	15	23	25	23	14
IBK1	% correctly labeled positive	34.10	55.50	62.70	65.70	66.50	67.60	67.80	68.00	68.00	68.20
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	28.30	44.53	47.11	45.93	42.96	40.62	37.68	34.99	32.34	30.09
IBK3	% correctly labeled positive	31.60		60.10	65.20	66.40	67.60	67.80	68.00	68.00	68.20
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	25.80	-10.97	44.51	45.43	42.86	40.62	37.68	34.99	32.34	30.09

	NORM Data Set									
		Times Positive Results Repeated									
		1	2	3	4	5	6	7	8	9	10
J48	% correctly labeled positive	29.50	45.80	52.60	56.10	60.80	65.10	65.50	66.80	67.00	67.30
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	23.70	34.83	37.01	36.33	37.26	38.12	35.38	33.79	31.34	29.19
	# of Leaves	41	64	77	94	114	109	114	118	119	123
	size of tree	81	127	153	187	227	217	227	235	237	245
JRIP	% correctly labeled positive	27.00	43.80	53.30	58.70	61.30	63.10	63.90	64.30	65.40	65.30
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	21.20	32.83	37.71	38.93	37.76	36.12	33.78	31.29	29.74	27.19
	# of Rules	8	6	16	19	23	17	21	11	16	20
IBK1	% correctly labeled positive	31.90	52.50	59.80	63.30	64.60	66.30	67.00	67.30	67.40	67.70
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	26.10	41.53	44.21	43.53	41.06	39.32	36.88	34.29	31.74	29.59
IBK3	% correctly labeled positive	30.90	45.30	57.80	62.90	64.40	66.30	67.00	67.30	67.40	67.70
	GUESS	5.80	10.97	15.59	19.77	23.54	26.98	30.12	33.01	35.66	38.11
	DIFF	25.10	34.33	42.21	43.13	40.86	39.32	36.88	34.29	31.74	29.59

The learners used for classification include: J48 decision tree, Jrip rule-based learner, Instance based learner (KNN) with K = 1 and K = 3, and SVM. The SVM results were least successful and were not included in the reports. The group also discouraged the use of SVM because it did not produce a rule set that could be used in future software. Comparing the two tables, there is a slight increase in accuracy with the inclusion of the anonymity feature, and the accuracy improves as the oversampling increases. The instance-based learners combined with

anonymity inclusion scored the highest accuracy of 68.2%. 1NN and 3NN performed the same in both feature sets.

Group C trained models using their Ask.fm data set and evaluated their performance using accuracy, precision, recall, F1-measure, and F2-measure. The group put a high emphasis on recall, as the goal is to identify as much cyberbullying as possible, and therefore relied on F2-measures for the final evaluation. With 23684 instances (278 positive for bullying), the data was divided 70/30 for training and evaluation. Experiment 1 used LSF (lexical syntactic features), which has proven effective in previous research. This feature set relies on word dependencies generated by the Stanford Parser. The below capture displays the evaluation results:

Method	Accuracy	Precision	Recall	F_1-measure	F_2-measure
LSF	98.43%	36.19%	45.24%	40.21%	43.08%

The group attributed the model's poor performance to the data set's poor grammar and spelling, deeming the Stanford Parser basically ineffective as it could not recognize the internet vocabulary. The group's 2nd experiment used a combination of 4 features sets with a SVM classifier.

Feature Sets	Accuracy	Precision	Recall	F_1-measure	F_2-**measure**
A	97.33%	24.53%	61.90%	35.14%	47.44%
B	96.88%	23.68%	75.00%	36.00%	52.32%
B+C	98.09%	33.94%	66.67%	44.98%	55.89%
B+C+D	97.62%	28.78%	70.24%	40.83%	54.53%

Clearly accuracy is not a good measure, as the models can achieve roughly 99% accuracy by predicting every instance is negative. With a focus on recall, the group uses the F2-measure to compare models. All models from experiment 2 outperformed experiment 1. The SVM performed best with feature sets B and C. Recall that set B was a modified bag of words model,

which used the baseline TFIDF scores (A) and multiplied offensive words by a scalar, increasing their weight. Set C introduced counts for offensive words and pronouns. It makes sense that inclusion of pronouns would increase overall performance.

Group D used the largest feature set on their Twitter data set, taking many factors into consideration, thinking beyond the text and word relations and including user profile, interaction, and session features. Several types of decision trees were experimented with, including J48, LADTree, LMT, NBTree, Functional Tree, and Random Forests. The Random Forest, tuned to 10 tree ensemble, performed the best of all learners. To handle data imbalance, they use a combination of under sampling and over sampling, suggesting that this produces the best results. Results were best when spam was removed and when data imbalance was adjusted. The below figure displays these results, generated with the RF model:

Table 3: Features evaluation.

	Prec.	Rec.	ROC
bully	1	0.667	0.833
aggressive	0.5	0.4	0.757
normal	0.931	0.971	0.82
overall (avg.)	0.909	0.913	0.817

Table 4: Classification on balanced data.

The Random Forest model was able to effectively predict bullying, aggressive, and normal Twitter comments.

Group E used data from Twitter, ASK.fm, and Instagram along with their PVC model to detect cyberbullying in a weakly supervised manner. Provided with a seed of offensive words,

the PVC model identifies bully vs victim and additional words that are indicative of bullying. Their results are presented below:

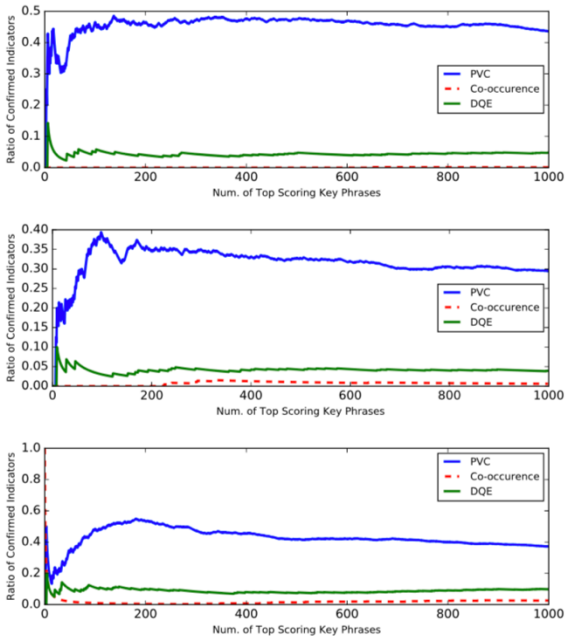

Fig. 2: Precision@k for bullying phrases on Ask.fm (top), Instagram (middle), and Twitter (bottom).

The PVC model is compared to baselines Co-occurrence and DQE, performing better than both and confirming that PVC is capable of detecting user roles in bullying and new words that are indicative of bullying.

Group F, using Twitter and YouTube data for training and a test data set provided by Kaggle, experimented with Logistic Regression, SVM, Random Forest, and Gradient Boosting models. Models are implemented in Python using the Scikit-Learn machine learning library. Their results are posted below:

the test accuracy lies between 70% - 75%.

Table 3. Accuracy score of train and test dataset.

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation
Logistic Regression	0.900	0.737	0.777	0.720
SVM	0.966	0.793	0.778	0.759
Random Forest	0.905	0.745	0.739	0.747
Gradient Boost	0.974	0.792	0.779	0.753

In general, the SVM and Gradient Boot models performed the best, with AUC scores on the test data set of .778 and .779, respectively. With high training accuracies near 97%, and low-test accuracies below 80%, it can be concluded that the models have nearly memorized their training data and may be suffering from overfitting.

The Support Vector Machine was a popular choice for classification, used by several of the groups. There was also a tendency to favor models that create a knowledge representation or rule set that can be understood, which is not possible with the SVM as it functions more as a black box. The Jrip rule-based learner and various decision trees were used to perform classification with success.

More recently there has been a lot of success in the field of NLP with deep neural networks, including RNN and LSTM networks as their recurrent feedback allows the model to be understand long distance dependencies, something very common in text input. A recent personal project deployed a LSTM network to perform sentiment analysis on movie reviews and achieved reasonable accuracy on the balanced test data set (> 85%). One requirement for deep neural networks is a large amount of data, as these networks typically generalize better and better as more data is used in training. This circles back around to the challenge of acquiring larger and better-quality data sets, a challenge that will lead to better classification results once overcome.

Conclusions

Cyberbullying was defined and presented as a major global problem that is affecting more than half of the teenage and young adult population. Several of the primary challenges associated with cyberbullying were introduced, including challenges native to natural language processing and challenges related to the acquisition of sufficient quality data. A total of 6 research studies in the area of cyberbullying detection with machine learning, spanning the years 2011 - 2018, were selected and examined to gain insight into the current state of the problem and its solutions. The machine learning process was divided into three categories: data collection and labelling, pre-processing and feature selection, and model training/evaluation. For each machine learning sub-task, all 6 research groups' methodologies were presented and compared to each other, providing the reader with multiple solutions to the problem at hand. Most research groups used a bag of words with weighted term frequency (TFIDF) for some portion of their study, unsurprisingly as TFIDF is a well-known NLP weighting method. The groups that spent additional time engineering features, including user based, network based, and word dependencies (e.g. pronouns), achieved better results on average. Finally, personal knowledge and experience were used to provide insight into possible future endeavors in the area of cyberbullying and machine learning.

References

1. Dinakar, K., Reichart, R., Liberman, H. (2011). “Modeling the Detection of Textual Cyberbullying”
2. Reynolds, K. (2011). “Using Machine Learning to Detect Cyberbullying”
3. Engman, L. (2016). “Automatic Detection of Cyberbullying on Socil Media”
4. Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G. (2017). “Mean Birds - Detecting Aggression and Bullying on Twitter”
5. Raisi, E., Huang, B. (2017). “Cyberbullying Detection with Weakly Supervised Machine Learning”
6. Sahay, K., Khaira, H., Kukreja, P., Shukla, N. (2018). “Detecting Cyberbullying and Aggression in Social Commentary Using NLP and Machine Learning”
7. Hosseinmardi, H., Mattson, S., Rafiq, R., Han, R., Lv, Q., Mishra, S. (2015). “Detection of Cyberbullying Incidents on the Instagram Social Network”
8. Pennington, J., Socher, R., Manning, C., (2014) “GloVe: Global Vectors for Word Representation”
9. Wolfe, L, (2017). “Twitter User Statistics 2008 Through 2017”
10. (2010). “Cyberbullying: A National Epidemic”, https://study.com