Detecting Cyberbullying: Query Terms and Techniques

April Kontostathis
Ursinus College
Collegeville PA
akontostathis@ursinus.edu

Kelly Reynolds
Lehigh University
Bethlehem PA
ker212@lehigh.edu

Andy Garron
University of Maryland
College Park, MD
agarron@cs.umd.edu

Lynne Edwards
Ursinus College
Collegeville PA
ledwards@ursinus.edu

ABSTRACT

In this paper we describe a close analysis of the language used in cyberbullying. We take as our corpus a collection of posts from Formspring.me. Formspring.me is a social networking site where users can ask questions of other users. It appeals primarily to teens and young adults and the cyberbullying content on the site is dense; between 7% and 14% of the posts we have analyzed contain cyberbullying content.

The results presented in this article are two-fold. Our first experiments were designed to develop an understanding of both the specific words that are used by cyberbullies, and the context surrounding these words. We have identified the most commonly used cyberbullying terms, and have developed queries that can be used to detect cyberbullying content. Five of our queries achieve an average precision of 91.25% at rank 100.

In our second set of experiments we extended this work by using a supervised machine learning approach for detecting cyberbullying. The machine learning experiments identify additional terms that are consistent with cyberbullying content, and identified an additional querying technique that was able to accurately assign scores to posts from Formspring.me. The posts with the highest scores are shown to have a high density of cyberbullying content.

Author Keywords

Machine Learning; Cyberbullying Detection; Term Analysis; Latent Semantic Indexing.

ACM Classification Keywords

K.4.1. Public Policy Issues: Abuse and crime involving computers.

General Terms

Human Factors; Algorithms; Verification.

INTRODUCTION

Social networking sites are great tools for connecting with people. Posting links, sharing pictures and videos, creating groups, and creating events are all great ways to extend communication with peers. However, as social networking has become widespread, people are finding illegal and unethical ways to use these communities. In particular, we see that teens and young adults are finding new ways to bully one another over the Internet.

Patchin and Hinduja note that cyberbullying is a pervasive and important problem, particularly as more youths have unsupervised access to the Internet in general, and social networking sites in particular [11]. Willard defines cyberbullying as "willful and repeated harm inflicted through the medium of electronic text." It takes on many forms, including flaming, trolling, cyberstalking, denigration, harassment, masquerade, flooding, exclusion, and outing [16]. The goal of this work is to find ways to detect instances of cyberbullying by creating a language model based on the text in online posts.

In previous work we have identified Formspring.me as a website that provides a rich source of data for studying cyberbullying [13]. Furthermore, we were able to identify linguistic features that were used to train machine learning algorithms to detect cyberbullying [13]. The current work extends our previous work by identifying specific terms that are indicative of cyberbullying, and using these terms to generate queries to detect cyberbullying content.

The first half of this paper presents a set of experiments that we refer to as our "bag-of-words" language model. We believe there are multiple types of cyberbullying and no one query will be able to capture everything we would like to find. Therefore we analyze and test multiple queries to identify the most success combinations of these terms for use in automated detection algorithms.

The second half of the article presents the use of a supervised learning methodology to identify cyberbullying instances in an independent test set. These experiments identify additional terms that are indicative of cyberbullying, and also identify an additional successful querying technique for use in detection algorithms.

Thus we have two important contributions to this pervasive problem and timely research topic. First, we are able to successfully detect cyberbullying content, an outcome that has eluded the few researchers who are tackling this problem. Second, we have identified specific terms that are indicative of cyberbullying.

WebSci'13, May 2–4, 2013, Paris, France.

Related Work

Very few other research teams are working on the detection of cyberbullying. A misbehavior detection task was offered by the organizers of CAW 2.0, but only one submission was received. Yin, et.al determined that a baseline text mining system (using a bag-of-words approach) was significantly improved by including sentiment and contextual features. Even with the combined model, a support vector machine learner could only produce a recall level of 61.9% [18].

A recent paper describes similar work that is being conducted at Massachusetts Institute of Technology. The research is aimed at detecting cyberbullying through textual context in YouTube video comments. The first level of classification is to determine if the comment is in a range of sensitive topics such as sexuality, race/culture, intelligence, and physical attributes. The second level is determining what topic. The overall success of this experiment reached 66.7% accuracy. This project also used a support vector machine learner [6]. The data used in these experiments has not been publically released.

In Xu, et. al the authors describe their use of a variety of natural language processing techniques to identify bullying traces (a new concept that refers to online references that can be bullying instances themselves or online references that refer to offline instances of bullying). They use sentiment analysis features to identify bullying roles and Latent Dirichlet Analysis to identify topics/themes. The paper is designed to set baselines for a variety of tasks pertaining to bullying detection and includes a call for other researchers to improve upon these baseline techniques [17].

DATASET DEVELOPMENT

In this section we describe the collection and labeling of the data we used in our experiments.

Dataset Origin

The website Formspring.me is a question-and-answer based website where users openly invite others to ask and answer questions. What makes this site especially prone to cyberbullying is the option for anonymity. Formspring.me allows users to post questions anonymously to any other user's page. Some instances of bullying found on Formspring.me include: "*Q: Your face is nasty. A: your just jealous*" and "*Q: youre one of the ugliest bitches Ive ever fucking seen. A: have you seen your face lately because if you had you wouldn't be talkin hun (:.*" It is interesting to note that the reactionary tone of the answers lends weight to the labeling of these interactions as containing bullying content. As noted in [17] sometimes bullying is identified by a defensive or aggressive response.

To obtain this data, we crawled a subset of the Formspring.me site and extracted information from the pages of 18,554 users. The XML files that were created from the crawl ranged in size from 1 post to over 1000 posts. For each user we collected the following profile information: date the page was created, userID, name, link(s) to other sites, location, and biography.

The name, links and biography data were manually entered by the user who created the page (the Formspring.me account) so we cannot verify the validity of the information in those fields. In addition to the profile information, we collected the following information from each Question/Answer interaction: Asker UserID, Asker Formspring.me page, Question, and Answer.

Labeling the Data

We extracted the question text and the answer text from a randomly chosen subset of the Formspring.me data and used Amazon's Mechanical Turk service to determine the labels for our corpus. Mechanical Turk is an online marketplace that allows requestors to post tasks (called HITs) which are then completed by workers. The "turkers" are paid by the requestors per HIT completed. The process is anonymous (the requestor cannot identify the workers who answered a particular task unless the worker chooses to reveal him/herself). The amount offered per HIT is typically small. We paid three unique workers 5 cents to label each post. Each HIT we posted displayed a question and its corresponding answer from the Formspring.me crawl and a web form that requested the following information:

1. Does this post contain cyberbullying (Yes or No)?
2. On a scale of 1 (mild) to 10 (severe) how bad is the cyberbullying in this post (enter 0 for no cyberbullying)?
3. What words or phrases in the post(s) are indicative of the cyberbullying (enter n/a for no cyberbullying)?
4. Please enter any additional information you would like to share about this post.

The primary advantage to using Mechanical Turk is that it is quick. Our dataset was labeled within hours. For our first experiments, we asked three workers to label each post because the identification of cyberbullying is a subjective task. Our class labels were "yes" for a post containing cyberbullying and "no" for a post without cyberbullying. The data provided by the other questions will be used for future work. At least two of the three workers had to agree in order for a post to receive a final class label of "yes" in our training and testing sets.

In our first labeling run (used in our bag-of-words experiments, below), 1185 of the 10685 posts contained cyberbullying (11.1%).

LANGUAGE MODEL (BAG-OF-WORDS)

A bag-of-words language model was chosen for our first set of experiments because the model is simple and transparent. The terms used in cyberbullying posts will be ready available. In contrast, more complex algorithms, such as

support vector machines, produce outcomes without identifying the underlying characteristics of the model produced by the machine. Bag-of-words is a keyword-based vector-space model, and Baeza-Yates and Ribeiro-Neto assert that this model "recognizes that the use of binary weights is too limiting and proposes a framework in which partial matching is possible." We created a term-by-document matrix for the 10,685 posts in our training set, and then used this matrix to run queries. We then measured the effectiveness of each query for retrieving cyberbullying content from the collection [1].

We indexed our corpus in the traditional way, converting all characters to lowercase, and removing numerics and special characters. We did this fully recognizing that in online communication, case and special characters often provide clues about emotion and context. For example, typing a post in all capitals is used for emphasis, similar to raising one's voice in face-to-face conversation. In the same way, emoticons (sequences of special characters that represent little pictographs) are use to explicitly convey emotion (smiley face for kidding, frowny face for angry or sad). We plan to enhance our model with these features in the future.

Once we were left with lowercase text we did no further preprocessing, resulting in a term list that contained 15,713 unique words. Thus the size of our term-by-document matrix was 15,713 by 10,685. Most of the entries were zeroes, of course, so the matrix was stored in a sparse format. The nonzero entries represented simple term counts (number of times term *t* appeared in post *p*).

Language Model Methodology

One thing was clear from results provided by the labelers from Mechanical Turk (turkers): there are "bad" words that make a post more likely to be labeled as cyberbullying. In order to leverage this information, we identified a list of insult and swear words posted on the website www.noswearing.com.

For this article we used each of the terms collected from www.noswearing.com as single word queries for detecting cyberbullying content. From each query we captured statistics, including the number of results returned, and the number of these that were identified as containing cyberbullying content by the turkers (true positives). The queries returned non-negative scores for each post. Higher scoring posts had more matching words but the order of the words in the post was unimportant (which is why the model is referred to in the literature as a "bag-of-words"). Posts with a score of 0 were removed from the result list. We also capture information about the number of true positives that appeared in the top 10 results, the top 20, the top 30, the top 100, and the top 500 results. The intention is to find terms that generate a lot of true positives with the highest scores. These experiments were used to identify content words that can be used to detect cyberbullying.

We were also interested in context, and to capture context we used each post that contained cyberbullying content as a query. Thus we had 1185 context-based queries. We captured the same metrics in these experiments as we did in the content experiments.

For both the content-based and the context-based experiments, we ran an optimization loop to determine appropriate term cutoff thresholds. We used minimum thresholds of 1 and 2 and maximum thresholds of 100, 300, 500, 700 and 900. For example, if we were using a lower threshold of 2 and upper threshold of 500, we ignored any term that appeared less than twice or more than 500 times in the collection. These thresholds were more useful in the context-based experiments because more words were used to match on as we increased the upper threshold, and words were removed as we increased the lower threshold.

After we obtained and analyzed the results from our bad-word dictionary and full-post query runs, we were able to develop queries for cyberbullying detection. In the next section we describe these queries and the results obtained when we ran these queries on our corpus.

Language Model Results

Precision and recall are typically used to express the quality of an information retrieval system. Precision is defined as the percentage of retrieved documents which are relevant to the query. Recall is the percentage of all relevant documents that were retrieved.

These metrics can be applied in two ways. First, we can compute recall and precision at rank = *n*, where *n* is a constant. In this case, we look at the first *n* documents returned from the query and compute the precision and recall using the above definitions. Another metric, F1, is computed as 2*(precision*recall)/(precision+recall). It is the harmonic mean between the two values and measures overall effectiveness of a query [15].

Content-Based

Our bad words dictionary contains 296 terms, ranging from the common-place (*shithead*) to the esoteric (*unclefucker*). We ran each of these terms individually through our query processing system. Some of the terms are hyphenated (*ass-hole*) and these were split into two words by our preprocessor.

Evaluation of the results is a bit tricky. The results with the highest precision typically reach that level of precision because there are few occurrences of that term in the corpus and those occurrences are all in bullying posts. For example, the term *bitchass* appears in 4 posts. All 4 are labeled as containing cyberbullying, and therefore the precision is 1. However the recall for this post is only .003. This term clearly won't be helpful for pulling all of the cyberbullying information from the collection. There are 15 terms of this type.

Similarly, there are another 15 terms that retrieve only a few posts, and these posts do not contain cyberbullying content. These 15 queries have both recall and precision equal to 0. Some of these are surprising, for example, *piss*, *jackass* and *bastard* fall into this category.

There are 176 terms that do not appear in our corpus at all. These all return a recall of 0 and the precision is undefined. Some of them would indicate cyberbullying should they appear (*dyke*, *beaner*), so we did not want to eliminate them because our corpus is small and they may be useful on a larger corpus.

ContentQuery1	Bitchass skank dickhead friendless fuckoff fuckstick kunt assfuck clitface dumshit faggottt fuckface motherfucking negro tard cunt fag hoe fuckin nasty bitch douche-fag faggot nigger trash
ContentQuery2	ass-shit, ass-fuck, ass-face, shit, bitch, fuck, ugly, ass-bite, ass-fucker, ass-cock, ass-hole, ass-hat, ass-nigger, ass-monkey, ass-clown, ass-pirate, ass-sucker, ass-wipe, ass, ass-banger, ass-cracker, ass-hopper, ass-jaber,ass-jacker, ass-licker, ass-wad, fucking, big, fake, gay, dick, stupid, hoe, pussy, damn, hell, dumb, fat, kill
ContentQuery3	Bitch, fucking, hoe, ugly, cunt, fag, ass-shit, pussy, Dumbass, douche-fag, nigger, trash, ass-fuck, shit, ass-fucker, ass-cock, ass-nigger, ass-monkey, ass-clown, ass-pirate, ass-sucker, ass-wipe, ass, ass-banger, ass-cracker, ass-hopper, ass-jaber, ass-jacker, ass-licker, ass-wad, fake, dick, damn, fat, whore, fuckin, stfu, fucked, faggot, fuck, ass-bite, ass-hole, kill, nigga, bitches, fucks, loser, dicks
ContentQuery4	All terms in the bad words dictionary.

Table 1. Content-based Queries.

In order to develop cyberbullying queries we looked at three criteria. We first looked at terms that produced recall levels that are in the top quartile for precision (precision >.75). There are 25 terms that meet these criteria (see Table 1, ContentQuery1). We used a lower bound of 1 and an upper bound of 900 to develop the queries. The single word queries based on the bad word dictionary were not greatly affected by the term count parameter, however, except when the upper bound got too low. For example, *ass* appears 226 times in our corpus, so queries that included the word *ass* were not successful when the upper bound was 100.

For our second content-based query we looked at the words that resulted in the highest recall. The words that make up this query appear as ContentQuery2 in Table 1. This query contains all of the terms with a true positive count greater than or equal to 50. There are 39 terms in the query. Because our system will break hyphenated terms into two words, we removed duplicates in order to prevent heavy weighting by independent components of compound terms (i.e. *ass* appeared only once in the final version of this query).

The third query (ContentQuery3) was based on the precision at rank 10. We are most interested in terms that produce good results at top ranks. Therefore, we collected all terms that had at least 5 posts in the top 10 that were true positives. There are 48 terms on this list. Once again duplicates of compound components were removed. Finally we ran a query containing all of the terms in the bad word dictionary, just for comparison purposes (ContentQuery4).

Context-Based

We ran the context-based queries and looked at the output metrics in similar ways. There are 30 queries in this group with precision of 1 (some are duplicates at different term thresholds). The term thresholds tell an important story here. Although we used thresholds up to 900, almost all of the precision 1 queries used an upper threshold of 100. This is to be expected, as we pull in more helper words, we will find these words in posts that do not contain cyberbullying content and therefore we expect the overall precision to decrease. There did not seem to be any advantage in building a query from the high precision posts, because the highest number of true positives returned was only 3.

Filtering for our largest term count range (lower bound=1, upper bound = 900), we identified the posts that receive an overall precision in the top quartile. There are 7 of these posts (see Table 2, duplicates were removed), and they appear to be promising, because they cumulatively returned 1313 true positives, many of them at top ranks. Some duplicates were obviously returned because there are only 1185 true positives in the entire collection. We combined these 7 posts into our first context-based query (contextQuery1). When processing this query, we did not eliminate duplicate terms, as a result terms like *shove* or *bitch* that appear twice will be given a higher weight. Keeping the duplicates provides additional information about context (words that are often repeated are given higher weight or authority).

Our second context-based query was comprised of the top 10 queries with the highest number of true positives returned. These queries are fairly long, and the average precision for this set of 10 posts is fairly low (.16), because they also return a lot of non-cyberbullying posts. However, they each returned over 800 true positives, so we decided to collect them into one query as ContextQuery2.

Finally, we used the 30 posts that had at least 9 true positives in the top ranked 10 documents as ContextQuery3 (duplicate posts were removed, but not duplicate words). This query appears to be more promising, because the average precision for these 30 posts is .51.

Query Results

The retrieval performance results from the 4 content-based and 3 context-based queries appear in Table 3. The precision and recall numbers were calculated using the best term range cutoffs for each metric (lower and upper bounds). As expected, overall recall benefitted from a wider range of values, and overall precision benefitted from a more narrow range. The best recall was produced by ContextQuery2, when the term range was allowed to grow to include terms with up to 900 occurrences in the corpus. This .95 recall, however, was accompanied by a low precision of .12, just barely better than a guess. The best precision was achieved by Content Query 1 at the narrowest term range (term counts between 1 and 100). The .84 precision was accompanied by a very low recall of only .09.

Query Text	Num True Positives	TP in top 10	Precision
Q: whooorrreeeeee A: No liifffeeee	1	1	1.00
Q: hoe A: thankssssssssssssssss	99	9	0.85
Q: shove it/ A: shove what bitch	260	9	0.79
Q: NIGGER A: sup bitch	260	9	0.78
Q: your a hoe A: anonomous	104	9	0.77
Q: **bitch*** lol VVV A: .	268	9	0.76
Q: bitch you kant rapp niqquh.! A: suk my dik pussy	321	9	0.76
Total	**1313**		

Table 2: Queries with Precision in Top Quartile (ContextQuery1)

Query	Precision	Recall
ContentQuery1	.84	.37
ContentQuery2	.50	.69
ContentQuery3	.56	.68
ContentQuery4	.44	.78
ContextQuery1	.72	.33
ContextQuery2	.18	.95
ContextQuery3	.25	.88

Table 3: Best precision and recall for each query

The best results overall were achieved by ContentQuery3, followed closely by ContentQuery2. Table 4 shows the best F1 score for each query, and the range cutoffs that achieved this score.

The lower bound does not seem to affect the queries much, but the optimal upper bound varies quite a bit between the content- and context-based queries. The long context-based queries (ContextQuery2 and ContextQuery3) had better performance at very low cutoff values. This result is consistent with the higher retrieval performance numbers produced by the content-based queries. The context-based queries are using low cutoffs to prune out filler words as much as possible to improve performance. Thus, we are not getting as much "context" as we expected to from these queries.

In is important to note, however that ContentQuery1, ContentQuery2, ContentQuery3, ContextQuery1 and ContextQuery3 all achieved very high precision at top ranks. The average precision at rank 100 for these queries was 91.25, with ContentQuery2 and ContextQuery3 providing the best results (.94). This high precision was largely impervious to changes in the term thresholds.

Query	F1	Lower	Upper
ContentQuery1	.49	1	500,700,900
ContentQuery2	.56	1,2	500,700,900
ContentQuery3	.57	1,2	500,700,900
ContentQuery4	.54	1,2	500,700,900
ContextQuery1	.45	1,2	500,700,900
ContextQuery2	.28	1,2	100
ContextQuery3	.31	1,2	100,300

Table 4. Best F1 for each query, and associated term count cutoff values.

Discussion

Our results are surprising and further investigation is warranted. Although we expected context to play an important role, the queries that just consisted of multiple "bad" words significantly outperformed the queries that included helper and filler words. However, both the content-based and the context-based queries produced high precision at top ranks. Moreover, the longer content-based queries were most effective, but use of the full bad-words dictionary hurt performance. An analysis of the query terms in ContentQuery2 and ContentQuery3 shows that these are typical insulting words used by teens and tweens. Terms such as: *bitch*, *whore*, *hoe*, *pussy*, *gay*, *ass*, *loser*, *nigga*, etc.

MACHINE LEARNING APPROACH (EDLSI)

In this section we use a supervised machine learning called Essential Dimensions of LSI (EDLSI) approach to identify additional terms and another approach to querying for the detection cyberbullying in our Formspring.me data.

Background

Baeza-Yates and Ribeiro-Neto describe the standard vector space retrieval process as follows: using vector space retrieval, documents are represented as vectors of dimension m x 1, where m is the count of terms in the dataset and position [i; 1] of each vector represents how

many times term *i* appears in the document [1]. Queries are represented in the same way (vectors of term-counts), and it is assumed that documents with vectors closer to the query vector are more relevant to the query. Cosine similarity is traditionally used as a metric to measure vector distance.

One limitation of standard vector space retrieval is that if none of the query terms appear in a document, that document will not be returned as relevant. Latent Semantic Indexing (LSI) has been shown to use second-order and higher-order word co-occurrence to overcome synonymy and polysemy problems in some corpora [8].

LSI is an extension of the generalized vector space model. It is designed to extract the meaning of words by using their co-occurrences with other words that appear in the documents of a corpus [4]. Latent Semantic Indexing, like Vector Space retrieval, uses the term-document matrix of a dataset, i.e. the matrix that contains a dataset's terms in its rows and documents in its columns, with position [*i; j*] representing how many times term *i* appears in document *j*.

LSI uses a linear algebraic technique known Singular Value Decomposition (SVD) to reduce noise in the term-document matrix. The SVD is a matrix factorization method that splits a matrix, **A**, into three matrices **U**, **S**, and **V**. The power of LSI comes from truncating the **U**, **S**, and **V** matrices to *k* dimensions. The truncated SVD produces the best rank-*k* approximation of the original term-document matrix [3]. In practice, fast algorithms are used to compute the partial SVD to *k* dimensions [5].

One of the disadvantages of LSI is that *k* (how far we calculate the dimensions of the SVD matrices) must be large enough to capture the latent semantic information for the corpus of interest. It has been shown that if only a few dimensions of the SVD are computed, they contain the most important data that the LSI extracts from the dataset, but not enough to accurately run searches [7]. However, these Essential Dimensions of LSI (EDLSI) can still be used, in conjunction with the original term-document matrix, to run queries effectively. In EDLSI we use the LSI scores on documents to try and draw out the latent semantics in a dataset, while also using the raw power of vector-space retrieval, by simply producing a weighted average of the results from LSI and the results from vector-space. In practice, a weighting parameter of .2 (where 20% of the contribution comes from LSI and 80% from traditional vector space retrieval) has been shown to produce good results on a variety of collections.

Machine Learning Methodology

Our training set for these experiments consisted of 13,652 Formspring.me posts that were previously labeled using Mechanical Turk as described above. Our testing set contained 10,482 unjudged posts. We discuss our labeling of the test instances in the results section, below. The entire collection of 24,134 posts was indexed together for input to the EDLSI processor.

The users of Formspring.me use typical Internet communication styles, particularly those employed by teens and tweens. In this set of experiments, in order to reduce the size of the index and to normalize the data for optimal term matching, we applied standard term pruning processes. For example, we removed terms that contained any non-alphabetic character, and converted everything to lower case. As noted above in the bag-of-words experiments, these choices represent an important consideration in the online environment where capitalization and sequences of non-alphabetic characters are used to convey emotion and sentiment.

We decided to clean the text in this way because it was the quickest approach to getting starting with this important task. Our previous work on detecting cyberpredation, has shown that the use of netspeak features is sometimes, but not always helpful [9]. More work is needed to identify precisely when these translations should be applied.

In order to further reduce the index, as well as to account for some of the netspeak issues, we implemented a term compression algorithm. Repeated sequences of the same character within a word were removed. As an example, *oooohhhhhhhhhh* became *oh*, and *looooovvvveeeeee* became *love*. This also results in some noise added in because we end up with non-words (ex. *call* became *cal*); and words sometimes were compressed to a word that means something entirely different (ex. *good* became *god*). Overall, we believe the benefits of term compression outweighed the disadvantages, because it allowed EDLSI to form more co-occurrence connections within the corpus.

After term compression was applied we further pruned the term space by removing words that only appeared once or twice in the entire collection, and also removed words that contained only 1 character or more than 17 characters. The resulting space contained 6473 terms. After preprocessing, we used Lemur to index the data and produce the term-by-document matrix. The Lemur Toolkit is designed to take a set of documents and add them quickly to an index [10]. We also used Lemur to apply the Term Frequency-Inverse Document Frequency (tf-idf) weighting scheme to our corpus. Term frequency (tf) measures the local importance of a term within a particular document. Inverse document frequency (idf) measures the discriminatory power of the term within the entire corpus [14]. A package called "irlba" for the R language [2] was used to calculate the SVD of the corpus to 500 dimensions.

After the index was generated, and the SVD was produced, we developed a query based on the 916 posts in our training corpus that were labeled as containing cyberbullying. The process for creating the query was fairly simple. We took the document vector for each of the 916 posts and computed a vector average. The cosine similarity was computed between the query vector and the term-by-document matrix (vector space retrieval). The query vector was also projected into the LSI space, and the LSI query

was run. The EDLSI weighted average was then computed to give the final score for each post.

Analysis of the query itself is interesting. There are 2065 nonzero weightings. Most are very small, only 278 are greater than 1. Table 5 shows the term and query vector weight for the top 26 terms in the query. We see some surprising terms in this list. For example, we might expect the juvenile insult language that we found in our bag-of-words experiments, terms such as *hoe*, *gay*, *whore*, *stupid*, etc., but we are little surprised to see terms like *thanks*, *she*, *wtf*, *dnt* (don't) and *yo*. It appears as if some of these terms are reactions to bullying attempts (*thanks*, *wtf*), and others are defense mechanisms (*dnt*, *stop*). In [17] these terms are considered "bullying traces," indications of a response to a cyberbully.

Term	Value	Term	Value
Hoe	15.0	Yo	5.7
Bitch	13.1	Her	5.6
Gay	12.0	Fuk	5.4
Fake	10.6	suck	5.3
Ugly	10.0	stop	4.9
Fuck	9.4	life	4.7
Pusy	7.3	fucking	4.7
Shit	6.9	okay	4.3
dick	6.8	ashole	4.2
wtf	6.5	thanks	4.0
face	6.1	whore	4.0
She	6.0	dnt	3.8
Anonomous	5.7	stupid	3.8

Table 5. Thirty top weighted query terms.

Machine Learning Results

After we ran the query and received the ranked posts as a result set, we filtered the set to remove all posts that we already had judgments for. In other words, although we indexed 24,134 posts to create our term-by-document matrix (and it is necessary to have it as one matrix for the EDLSI processing) we were really interested in the ranking of the 10,482 test instances in our result list.

In addition to returning a ranked list of posts, our system also assigns a score to each instance. A high score indicates a higher probability that a post contains cyberbullying content.

We labeled the training set by sending the full set to Amazon's Mechanical Turk and asking for judgments from three "turkers." This process is efficient and effective. It also can be costly when you have large sets of data to label (the cost for labeling 13,652 posts was $2047.80, not including the processing fee charged by Amazon). Therefore we took a slightly different approach when labeling the test data.

Approximately 7% of our training set consisted of posts that had cyberbullying content. We decided to send the 1000 question-and-answer pairs in our test set that received the highest scores to Mechanical Turk for labeling (9.5% of our test data). This number ensures that the percentage sent was greater than the density of cyberbullying in the training set, and also feasible from a cost perspective.

Of these top 1000 ranked documents, 465 were judged as containing cyberbullying (46.5%). This is a much higher density than we found in the training set, so we are confident that our EDLSI system is giving higher scores to posts that contain cyberbullying content.

Figure 1 shows the distribution of the true positives within the top 1000 posts. This is precisely the distribution that we want. Of the 83% of the posts with the 100 highest scores contain cyberbullying content and the density goes down in a linear fashion as the score decreases. Clearly the bullying content is being pushed to the top of our search results for this query. In fact, the average precision for the top 1000 documents with the highest scores is 67.1% (significantly higher than the 46.5% baseline for the top 1000 documents). Average precision is a standard information retrieval metric [15]. The 46.5% precision at rank 1000 is already a vast improvement over the 6.7% baseline provided by the training set; thus we have both visual and statistical evidence for the efficacy of our system.

Although we were happy with our results for the top 1000 documents we were also interested in seeing what we were missing. The cost to label the remaining 9482 was over $1400 (not including the $150 we spent to label the first 1000). We decided to have the remaining instances labeled by only one turker. This reduces the cost considerably and also gives us some indication of the number of true positives we might be missing. Of 9482 posts, 899 were given a "yes" label. Each of these was sent for further review by another turker. If the second turker also assigned a "yes", we stopped the process and assigned a final label of "yes" to the post. If the second turker labeled a post as "no," that post was sent for a third review. After this process completed we had 447 (down from 899) documents that were positive for cyberbullying. In total we sent another 1528 posts for further review. The total cost for labeling the remaining 9482 was $550.55. With these results we have identified 912 cyberbullying instances in total; thus, 8.7% of the dataset is known to contain cyberbullying content.

Figure 2 shows the number of cyberbullying posts by rank for the entire test set. We are happy to see that the trend is continuing. There is clearly a higher density of cyberbullying posts when the scores are higher.

Figure 3 presents the corresponding precision and recall curves. They are consistent with our findings. The overall

average precision across all documents is 47.7%, a significant improvement over the 8.7% baseline.

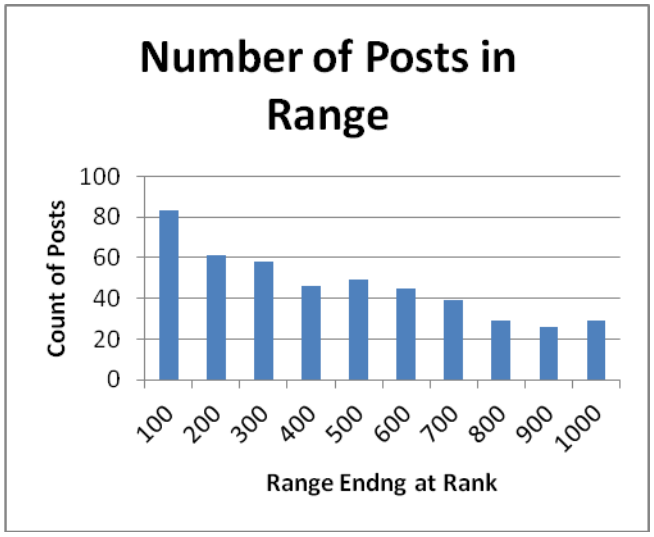

Figure 1. Number of cyberbullying posts in each range of 100 documents for the top 1000 ranked results.

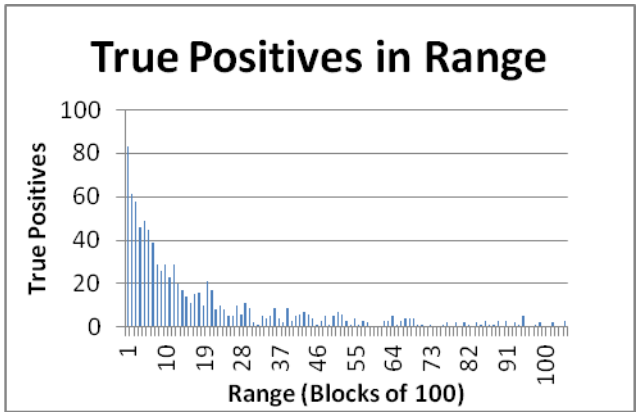

Figure 2. Number of cyberbullying posts in each range of 100 documents for all documents in the test set.

Machine Learning Discussion

We know we are missing some of the cyberbullying documents that are contained in the rank 1001 through rank 10482 results. Thus far we know that 447 of these are definitely cyberbullying, and an additional 986 are not cyberbullying. Information about the remaining 8584 posts is unknown. However, we can be confident about a few things pertaining to these 8584 posts. First of all, we know that at least one labeler indicated that each post was not a cyberbullying post. At a minimum we can conclude that any cyberbullying in the post is milder and perhaps even a borderline case (where some humans would consider it to be cyberbullying and some would not). Furthermore, we know from previous experience with other subsets from Formspring.me that the percentage of cyberbullying we find in any particular subset tends to range from 7-14%. Therefore, our 8.7% is consistent with other datasets. We also know that of posts that are labeled as cyberbullying approximately 52% are given "Yes" labels by all three turkers. These are stronger samples of cyberbullying. For the 465 positives in the top 1000, 272 resulted in three "Yes" votes (58.5%). Although we are certainly missing some true positives, we feel confident that our methodology represents a fair balance between learning as much as we can about the language used for cyberbullying, while at the same time using our limited resources wisely.

Our intention is to rebuild the query using the additional true positives and produce another test set to send for labeling. Table 6 shows the top 26 query terms and the weights when we add the new 912 true positives to our query (resulting in a query that was built from 1828 bullying samples). Overall the weights are higher, and some of the insult words have been moved to the top of the list. Interestingly, we still see some of the reaction words (*thanks*, *wtf*) and personal pronouns (*she*, *youre*) on the list.

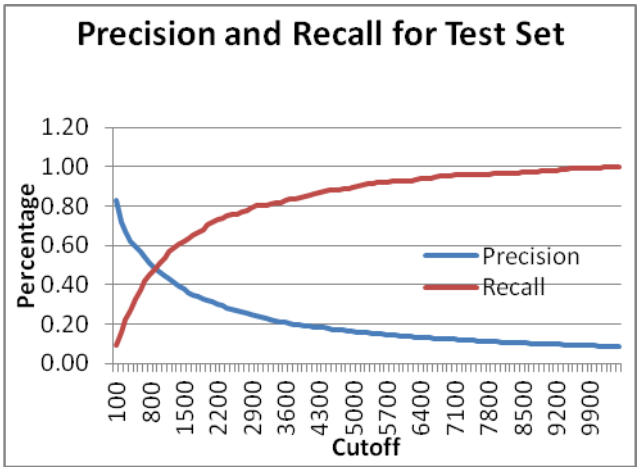

Figure 3. Number of cyberbullying posts in each range of 100 documents for all documents in the test set.

The cyberbullying instances in the lower ranks tend to be distinctly different from those at the top ranks. Figure 4 shows the text for the 10 true positive posts that were ranked lowest by our system. Contrast this with the 5 top ranked cyberbullying posts in Figure 5. The highly ranked posts tend to use a greater density of foul language and insults that most would consider to be bullying. The posts in the bottom 10 are more esoteric. We believe that different types of queries may rank cyberbullying types differently. Further analysis is required to develop a theory for the best sequence of queries to use to move all of the bullying to the top of the ranked retrieval list.

CONCLUSION

We have completed experiments to determine the terms most commonly used in cyberbullying incidents on the Formspring.me social networking site. A multitude of individual terms and messages were evaluated. We identified terms that are most indicative of cyberbullying. Additionally, two methods of developing queries for the

detection of cyberbullying were described and shown to be effective.

We also have presented a new process for the labeling of test data. Our process is more economical than the traditional approach of labeling every instance in a corpus, and is still effective. As corpora grow in size, labeling every instance becomes impractical or impossible. Our method strikes a balance between cost and effectiveness.

In future work, we would like analyze other possibilities, including finding other "bad" words that may prove helpful, and checking other types of words that might prove useful (McGhee, et.al. [9] found that personal pronouns were helpful for predation detection, for example). We would also like to normalize both for document length (posts range from 1 or 2 words to hundreds of words), and also for netspeak variations (both in words and phrases). Including information pertaining to emoticon usage would also be interesting.

Term	Value	Term	Value
whore	33.7	Wtf	7.3
hoe	15.1	Skank	7.3
bitch	13.8	Pusy	7.3
gay	12.0	Her	7.0
fuck	12.0	Face	6.9
ugly	11.0	Yo	6.4
fake	10.6	Stop	6.4
slut	9.9	Fat	6.4
youre	9.3	She	6.3
thanks	8.7	ashole	6.3
dick	7.9	suck	6.1
shit	7.8	fucking	6.1
stfu	7.4	hate	6.1

Table 6. Thirty top weighted query terms for next iteration.

1. q: are u a lezbo a: no
2. q: Lauren Jayme Olivia Taylor Kira Celia a: i dont carer i dont carer i dont carer do you have a lifer i guess not
3. q: u r meaty a: explain lol
4. q: chink a: sweet
5. q: Is it just me or do you not have eyebrows a: I have blonde eyebrows
6. q: prude a: Not
7. q: can i pop ur cherry a: lol
8. q: you are a butterface a: you
9. q: MIKE BRAY NO LIKEY YOU a: k
10. q: quieres folar? a: no.

Figure 4. Lowest ranked bullying posts.

1. q: ugly hoe a: thanks
2. q: She wont get maried a: LMAO Trust me theres a whole line of guys waitin for her Stop hatin on her She aint ugly like you Just because your ugly dont mean she ugly She prety so bitch fuck of and also do us al a favor and GET A LIFE yeah realy thanks
3. q: uhm bitch why did you get a formspring if you dont want people to say shit anonymously a: you are dumb bitch shut the fuck up
4. q: ask the guys who already have a: um Hoe its not being a hoe i didnt say it was alot of people its actualy only ben if u think im a hoe thats on you idgaf bitch hahahaha
5. q: ew germaine is a bitch a: Your face is a bitch Shut the fuck up

Figure 5. Top ranked bullying posts.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 0916152. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

1. Baeza-Yates, R., and Berthier Ribeiro-Neto (2011). *Modern Information Retrieval: The Concepts and Technology behind Search*. New York: Addison Wesley. Print.
2. Baglama, J. and L. Reichel (2011). irlba. http://illposed.net/irlba.html
3. Berry M.W, S.T. Dumais, and G.W. O'Brien (1995). Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):575-595.
4. Deerwester, S.C., S. T. Dumais, T. K. Landauer, G.W. Furnas, and R. A. Harshman (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41(6):391-407.
5. Demmel, J.W., J. Dongarra, B. Parlett, W. Kahan, D. Bindel, Y. Hida, X. Li, O. Marques, E.J. Riedy, C. Vmel, J. Langou, P. Luszczek, J. Kurzak, A. Buttari, J. Langou, and S. Tomov (2007). Prospectus for the next LA-PACK and ScaLAPACK libraries. *In Proceedings of the 8th international Conference on Applied Parallel Computing: State of the Art in Scientific Computing* (Ume, Sweden). B. Kgstrm, E. Elmroth, J. Dongarra, and J. Wasniewski, Eds. Lecture Notes In Computer Science. Springer-Verlag, Berlin, Heidelberg, 11-23.
6. Dinakar, K; Reichart, R.; Lieberman, H. (2011). *Modeling the Detection of Textual Cyberbullying*. Thesis. Massachusetts Institute of Technology.
7. Kontostathis, A. (2007). Essential dimensions of latent semantic indexing (LSI). *Proceedings of the 40th Hawaii International Conference on System Sciences*. January 2007.
8. Kontostathis, A. and W.M. Pottenger. (2006). A frame-work for understanding LSI performance. *Information Processing and Management.* Volume 42, number 1, pages 56-73.

9. McGhee, I., J. Bayzick, A. Kontostathis, L. Edwards, A. McBride, and E. Jakubowski. (2011). Learning to Identify Internet Sexual Predation. *International Journal on Electronic Commerce*. Volume 15, Number 3. Spring 2011.
10. Ogilvie, P. and J. Callan (2002). Experiments Using the Lemur Toolkit, In *Proceedings of the Tenth Text Retrieval Conference (TREC-10),* pages 103-108
11. Patchin, J. and S. Hinduja. (2006). Bullies move beyond the schoolyard; a preliminary look at cyberbullying. *Youth violence and juvenile justice*. 4:2,148-16
12. PC Magazine. (2011). Study: A Quarter of Parents Say Their Child Involved in Cyberbullying.(2011, July). *PC Magazine Online*. Academic OneFile. Web.
13. Reynolds, Kelly, April Kontostathis, and Lynne Edwards. 2011. Using Machine Learning to Detect Cyberbullying. In *Proceedings of the 2011 10th International Conference on Machine Learning and Applications Workshops (ICMLA 2011)*. December 2011. Honolulu, HI
14. Salton, G. and C. Buckley (1988). Term-weighting approaches in automatic text retrieval. *Information Process Management,* 24(5): 513-523
15. van Rijsbergen, C. J. (1979). Information Retrieval (2nd ed.). Butterworth.
16. Willard, N. E. (2007). *Cyberbullying and Cyberthreats: Responding to the Challenge of Online Social Aggression, Threats, and Distress.* Champaign, IL: Research. Print.
17. Xu, Jun-Ming; Kwang-Sung Jun; Xiaojin Zhu; and Amy Bellmore. Learning from bullying traces in social media. In *Proceedings of the 2012 Conference of North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT),* Montreal, Canada, 2012, pp.656-666.
18. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards. (2009). Detection of Harassment on Web 2.0 in CAW 2.0 '09: *Proceedings of the 1st Content Analysis in Web 2.0 Workshop*, Madrid, Spain.