Weakly Supervised Cyberbullying Detection using Co-trained Ensembles of Embedding Models

Elaheh Raisi
Department of Computer Science
Virginia Tech
Email: elaheh@vt.edu

Bert Huang
Department of Computer Science
Virginia Tech
Email: bhuang@vt.edu

***Abstract*—Social media has become an inevitable part of individuals' personal and business lives. Its benefits come with various negative consequences. One major concern is the prevalence of detrimental online behavior on social media, such as online harassment and cyberbullying. In this study, we aim to address the computational challenges associated with harassment detection in social media by developing a machine learning framework with three distinguishing characteristics. (1) It uses minimal supervision in the form of expert-provided key phrases that are indicative of bullying or non-bullying. (2) It detects harassment with an ensemble of two learners that co-train one another; one learner examines the language content in the message, and the other learner considers the social structure. (3) It incorporates distributed word and graph-node representations by training nonlinear deep models. The model is trained by optimizing an objective function that balances a co-training loss with a weak-supervision loss. We evaluate the effectiveness of our approach using post-hoc, crowdsourced annotation of Twitter, Ask.fm, and Instagram data, finding that our deep ensembles outperform previous non-deep methods for weakly supervised harassment detection.**

I. Introduction

The advent of social media has revolutionized human communication. Social media owes its increasing popularity to its numerous positive influences on individuals' social and business lives. It makes people closer to each other, provides access to enormous real-time information, and eases marketing and business. Despite these benefits, social media has amplified some detrimental aspects of society. Online harassment and cyberbullying are among the major adverse consequences of social media's popularity. According to the American Academy of Child & Adolescent Psychiatry [1], victims of bullying can suffer from issues in social and emotional development and can even be drawn to extreme behavior such as attempted suicide. Any widespread bullying enabled by technology represents a serious social health threat.

In this paper, we describe a new machine learning approach for harassment-based cyberbullying detection. We approach the cyberbullying detection problem from a different angle than many previously proposed machine learning methods. Most machine learning methods for this problem consider supervised cyberbullying detection, classifying social media posts as "bullying" or "non-bullying." In these approaches, humans annotate the data, and then a supervised classifier is applied to classify the posts. These methods rely on extracting textual and social features, then training a supervised classifier. There are, however, several challenges related to these approaches. Fully annotating data requires fine-grained human intervention, which is costly and time consuming. And without considering social context, differentiating bullying from less harmful behavior is difficult due to complexities underlying cyberbullying and related behavior. Our approach aims to encode such complexities into an efficiently learnable model without explicitly extracting features from data.

We use machine learning with *weak supervision*, which significantly alleviates the need for human experts to perform tedious data annotation. Our weak supervision is in the form of expert-provided key phrases that are indicative of bullying or non-bullying.

Our proposed *co-trained ensemble* framework consists of two learning algorithms that co-train one another, seeking consensus on whether examples in unlabeled data are cases of cyberbullying or not. One detector identifies bullying by examining the language content of messages; another detector considers the social structure to detect bullying. Training different learners on different perspectives of the problem aligns with the true multi-faceted nature of cyberbullying. Moreover, since the true underlying cyberbullying phenomenon is both linguistic and social, we should expect good models using each of these views to agree with each other, motivating our search for a consistency across the two perspectives.

We represent the language and users as vectors of real numbers with embedding models. For example, doc2vec [8] is a popular word-embedding model that represents documents with low-dimensional vectors (based on ideas from the word2vec per-word embedding [9], [10]). And node2vec [11] is a framework for building continuous feature representations for nodes in networks. We use language and user vectors as the input to language-based and user-based classifiers, respectively. We examine two strategies when incorporating vector representations of language. First, we use existing doc2vec [8] models as inputs to the learners. Second, we create new embedding models specifically geared for our specific task of harassment detection, which we train in an end-to-end manner during optimization of the model, incorporating the unsupervised doc2vec loss function into our co-training objective.

To train the model, we construct an optimization problem

made up of a regularizer and two loss functions: a co-training loss that penalizes the disagreement between the deep language-based model and the deep user-based model, and a weak-supervision loss that is the classification loss on weakly labeled messages.

We evaluate our approach on data from Ask.fm, Twitter, and Instagram, which are three of the public-facing social media platforms with a high frequency of cyberbullying. We use two human-curated lists of key phrases indicative and counter-indicative of bullying as the weak supervision, and we assess the precision of detections by variations of the framework. We evaluate the effectiveness of our approach using post-hoc, crowdsourced annotation of detected conversations from the social media data. We quantitatively demonstrate that our weakly supervised deep models improve precision over a previously explored, non-deep variation of the approach.

II. Related Work

Our framework is motivated by ideas studied in *multi-view learning* [2]–[5]. Multi-view learning is specifically useful when data is comprised of multiple views of some underlying phenomenon. Algorithms can exploit multi-view information to improve the learning performance. Blum and Mitchell [6] introduced a co-training method for multi-view learning, primarily for semi-supervised problems. These co-training algorithms alternately learn model parameters to maximize the mutual agreement across two distinct views of the unlabeled data. To understand the properties and behaviors of multi-view learning, some researchers have studied its generalization-error via PAC-Bayes and Rademacher complexity theory [7].

Many researchers have proposed computational methods for automated online harassment and cyberbullying detection. Most methods developed so far use supervised classification algorithms to classify messages as "bullying" or "non-bullying" by extracting language features. Some proposed gender-specific language features to classify users into male and female groups to improve the discrimination capacity of a classifier for cyberbullying detection [12]. Others applied a lexical syntactic feature (LSF) [13] approach to detect offensive content in social media and users who send offensive messages. Others focused on detecting textual cyberbullying in YouTube comments by manually labeling 4,500 YouTube comments and applying binary and multi-class classifiers [14].

Researchers have proposed methods that model posts written by bullies, victims, and bystanders using linear support vector machines and designing text-based features on an annotated corpus of English and Dutch [15]. Some researchers applied recurrent neural networks (RNN) by incorporating features associated with users' tendency towards racism or sexism with word frequency features on a labeled Twitter dataset [16]. Another approach used the number, density, and value of offensive words as features for cyberbullying identification on the Formspring service [17], [18]. There have been many contributions that design special features. For example, researchers designed features learned by topic models as well as curse words weighted by TF-IDF [19], sentiment features [20], features derived by applying vulgar language expansion using string similarity [21], features based on association rule techniques [22], and static, social-structure features [23]–[25]. Authors have used probabilistic fusion methods to combine social and text features together as the input of classifier [26].

Researchers have extracted text, user, and network-based attributes to study the behavior of bullies and which features distinguish them from regular users [27]. Similarly, some have extracted the properties of cyber-aggressors, their posts, and their difference from other users in the content of the *Gamergate controversy* [28], [29].

Other researchers extract a small set of social network structure features that are the most important for cyberbullying detection to improve the accuracy and time in an online setting [30], [31].

Researchers have also trained a supervised three-class classifiers using language features to separate tweets into three categories: those containing hate speech, only offensive language, and those with neither [32].

Additionally, some studies have involved firsthand accounts of young persons, yielding insights on new features for bullying detection and strategies for mitigation [33]. In another related direction, researchers introduced a user-centric view for hate speech, examining the difference between user activity patterns, the content disseminated between hateful and normal users, and network centrality measurements in the sampled graph [34].

Hosseinmardi et al. [35]–[38] conducted several studies analyzing cyberbullying on different online platforms, with findings that highlight cultural differences among the platforms.

While most studies on cyberbullying detection with machine learning focus on supervised learning, emerging approaches using weak supervision are beginning to appear [39], [40]. Our work directly builds on a recent paper that introduced the *participant-vocabulary consistency* (PVC) method [40], which uses a similar paradigm of viewing the learning tasks as seeking consensus between language-based and user-based perspectives of the problem. PVC uses simple key-phrase presence and a two-parameter user characterization, scoring how much a user tends to bully and how much they tend to be victimized, as its vocabulary and participant detectors, respectively. Our approaches replace these with richer classifiers.

A preliminary version of the study presented in this paper, including the methods and experiments, appear in our previous, non-archived workshop paper [41].

III. Co-Trained Ensembles

Our learning framework uses co-trained ensembles of weakly supervised detectors. In this section, we first describe them generally. Then we describe the specific instantiations we use in our experiments. A fundamental principle for our co-trained ensemble framework is the diversity of learners that look at the problem from different perspectives. Our framework trains two detectors; one detector identifies bullying incidents by examining the language content of messages; another detector considers social structure to discover bullying. To formally

describe social media data, we consider a set of users U and a set of messages M. Each message $m \in M$ is sent from user $s(m)$ to user $r(m)$. In other words, the lookup functions s and r return the sender and receiver, respectively, of their input message. The input data takes on this form, with some of the messages annotated with weak supervision.

A. General Framework

We define two types of classifiers for harassment detection: message classifiers and user-relationship classifiers (or *user classifiers* for short). Message classifiers take a single message as input and output a classification score for whether the message is an example of harassment, i.e., $f : M \mapsto \mathbb{R}$. User classifiers take an ordered pair of users as input and output a score indicating whether one user is harassing the other user, i.e., $g : U^2 \mapsto \mathbb{R}$. For message classifiers, our framework accommodates a generalized form of a weakly supervised loss function ℓ (which could be straightforwardly extended to also allow full or partial supervision). Let Θ be the model parameters for the combined ensemble of both classifiers. The training objective is

$$\min_{\Theta} \underbrace{\frac{1}{2|M|} \sum_{m \in M} \left(f(m;\Theta) - g\left(s(m), t(m);\Theta\right)\right)^2}_{\text{consistency loss}} + \underbrace{\frac{1}{|M|} \sum_{m \in M} \ell\left(f(m;\Theta)\right)}_{\text{weak supervision loss}},$$

where the first loss function is a consistency loss, and the second loss function is the weak supervision loss.

The consistency loss penalizes the disagreement between the scores output by the message classifier for each message and the user classifier for the sender and receiver of the message.

The weak supervision loss relies on annotated lists of key-phrases that are indicative or counter-indicative of harassment. For example, various swear words and slurs are common indicators of bullying, while positive-sentiment phrases such as "thanks" are counter-indicators. Let there be a set of indicator phrases and a set of counter-indicator phrases for harassment. The weak supervision loss ℓ is based on the fraction of indicators and counter-indicators in each message, so for a message containing $n(m)$ total key-phrases, let $n^+(m)$ denote the number of indicator phrases in message m and $n^-(m)$ denote the number of counter-indicator phrases in the message. We bound the message learner by the fraction of indicator and counter-indicator key-phrases in the message:

$$\underbrace{\frac{n^+(m)}{n(m)}}_{\text{Lower Bound}} < y_m < \underbrace{1 - \frac{n^-(m)}{n(m)}}_{\text{Upper Bound}},$$

If this bound is violated, we penalize our objective function using weak suppervision loss. The weak supervision loss is then

$$\ell(y_m) = -\log\left(\min\left\{1, 1 + (1 - \tfrac{n^-(m)}{n(m)}) - y_m\right\}\right) - \log\left(\min\left\{1, 1 + y_m - \tfrac{n^+(m)}{n(m)}\right\}\right).$$

This form of penalized bound is a generalization of the log-loss, or cross-entropy; in the rare cases that the weak supervision is completely confident in its labeling of a message being bullying or not, it reduces to exactly the log-loss.

B. Models

For the message classifiers, we use four learners:

(i) BoW: a randomly hashed bag-of-n-grams model with 1,000 hash functions [42],
(ii) doc2vec: a linear classifier based on the pre-trained doc2vec vector of messages trained on our dataset [8],
(iii) embedding: a custom-trained embedding model with each word represented with 100 dimensions,
(iv) RNN: a recurrent neural network—specifically a long-short term memory (LSTM) network—with two hidden layers of dimensionality 100.

The embedding and RNN models are trained end-to-end to optimize our overall loss function, and the vector-based models (BoW, doc2vec) are trained to only adjust the linear classifier weights given the fixed vector representations for each message.

For the user classifiers, we use a linear classifier on concatenated vector representations of the sender and receiver user nodes. To compute the user vector representations, we pre-train a node2vec [11] representation of the communication graph. Node2vec finds vector representations that organize nodes based on their network roles and communities they belong to. The pre-trained user vectors are then the input to a linear classifier that is trained during the learning process

There are eight combinations of message and user learners (including the option to use no user learner, in which case we are simply using the weak supervision loss to train the message classifiers). Figure 1 illustrates the model architecture for a co-trained ensemble of an RNN message learner and a node2vec user learner. The other possible combinations of message and user learners are analogously structured.

IV. Experiments

Mirroring the setup initially used to evaluate PVC [40], we construct our weak supervision signal by collecting a dictionary of 3,461 offensive key-phrases (unigrams and bigrams) [43]. We augment this with a list of positive opinion words collected by Hu & Liu [44]. The offensive phrases are our weak indicators and the positive words are our counter-indicators. We used three datasets in our experiments.

We use the **Twitter** data collected by Raisi & Huang [40]. They collected data from Twitter's public API, extracting tweets containing offensive-language words posted between November 1, 2015 and December 14, 2015. They then extracted conversations and reply chains that included these tweets. They then used snowball sampling to gather tweets in a wide range

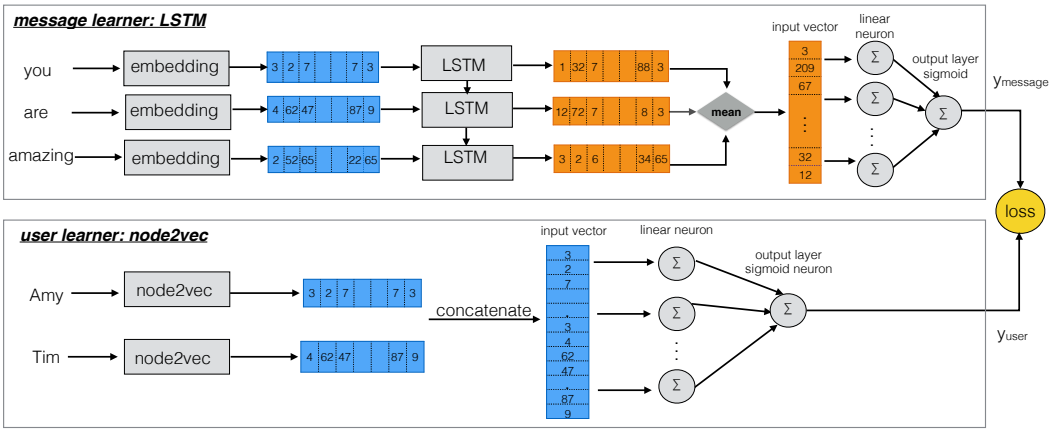

Fig. 1: Diagram of the co-trained ensemble of the RNN message learner and the node2vec user learner.

of topics. After some preprocessing, the Twitter data contains 180,355 users and 296,308 tweets.

We use a subsample of the **Ask.fm** dataset collected by Hosseinmardi et al. [36]. On Ask.fm, users can post questions on public profiles of other users, anonymously or with their identities revealed. The original data collection used snowball sampling, collecting user profile information and a complete list of answered questions. Since our model calculates the bully and victim scores for every user, it does not readily handle anonymous users, so we removed all the question-answer pairs where the identity of the question poster is hidden. Furthermore, we removed question-answer pairs where users only post the word "thanks" and nothing else, because this was extremely common and not informative to our study. Our filtered dataset contains 260,800 users and 2,863,801 question-answer pairs.

We also use a subsample of the **Instagram** dataset collected by Hosseinmardi et al. [37] via snowball sampling. For each user, they collected all the media the user shared, users who commented on the media, and the comments posted on the media. We filter the data to remove celebrities and public-figure accounts. Our filtered Instagram data contains 656,376 users and 1,656,236 messages.

A. Precision Analysis

We use post-hoc human annotation to measure how well the outputs of the algorithms agree with annotator opinions about bullying. We asked crowdsourcing workers from Amazon Mechanical Turk to evaluate the cyberbullying interactions discovered by all the methods. First, we averaged the user and message classification scores of each message. Then, we extracted the 100 messages most indicated to be bullying by each method. Finally, we collected the full set of messages sent between the sender and receiver of these messages. We showed the annotators the anonymized conversations and asked them, "Do you think either user 1 or user 2 is harassing the other?" The annotators indicated either "yes," "no," or "uncertain." We collected five annotations per conversation.

In Fig. 3, we plot the precision@k of the top 100 interactions for all the combinations of message and user detectors. We compare these methods with each other and against PVC [40], and two other baselines: *seed-based* and *naive-participant*. The *seed-based* method computes an detection score as the concentration of seed words in the message. The *naive-participant* method computes bully and victim scores for users as the frequency of messages with seed words sent and received, respectively, by the user. The bully and victim scores are then added to a vocabulary score, which is again the concentration of seed words in the message. The precision@k is the proportion of the top k interactions returned by each method that the majority of annotators agreed seemed like bullying. For each of the five annotators, we score a positive response as +1, a negative response as -1, and an uncertain response as 0. We sum these annotation scores for each interaction, and we consider the interaction to be harassment if the score is greater than or equal to 3.

We consider the results by grouping them into different message-classifier types. We first examine the doc2vec message detector as shown in top left of Figs. 2 to 4. The combination of doc2vec message detectors with the node2vec user detector produces the best precision in all three datasets.

We next examine the BoW message detector in the top right of Figs. 2 to 4. On Ask.fm data, we observe the best precision is achieved by the BoW message detector combined with the node2vec user detector. Interestingly, BoW on Twitter by itself does quite well; its precision is very similar to BoW with the node2vec user learner. The trend, however, is different on Instagram; PVC's performance is better than all variations of BoW. One possible reason why BoW does not do well on the Instagram data may be because we have short and sparse messages in our Instagram data. We also found some conversations were not fully in English, but another language using English characters. This sparsity may cause some tokens to occur only once in the data, causing the bag-of-words representation to be unable to generalize.

The third message detector is the RNN model shown on the bottom left of Figs. 2 to 4. On Twitter, the RNN by itself and combined with the node2vec user learner achieve higher precision than others. On Ask.fm data, all of the RNN message

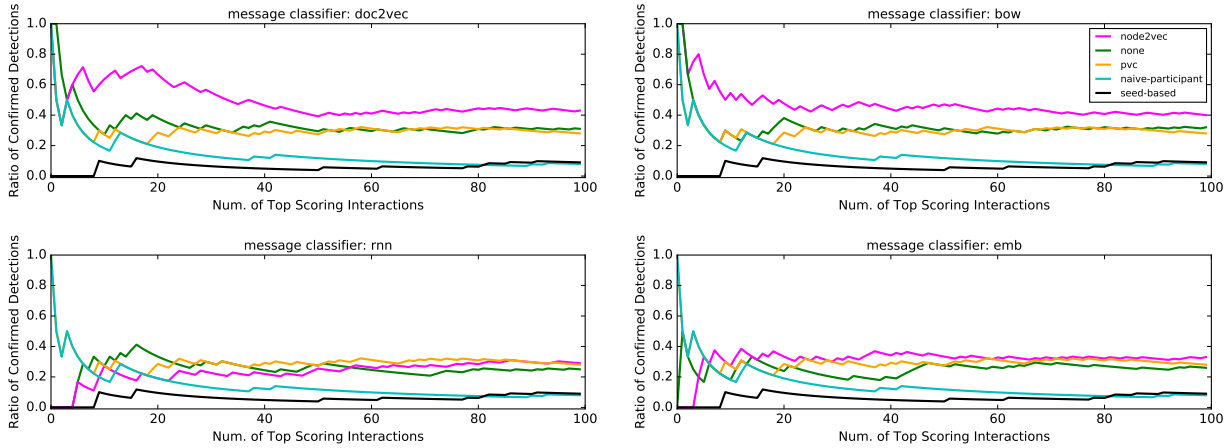

Fig. 2: Precision@k for bullying interactions on Ask.fm data using the combination of message and user learners, PVC, seed-based, and naive-participant.

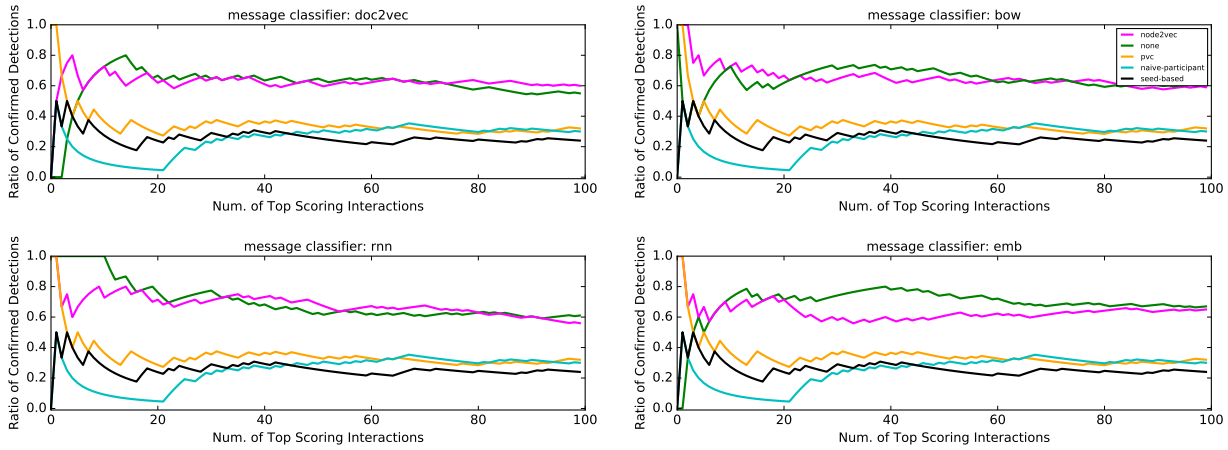

Fig. 3: Precision@k for bullying interactions on Twitter data using the combination of message and user learners, PVC, seed-based, and naive-participant. The legend in the top left plot applies for all other plots in this figure.

learner models behave similarly and slightly worse than PVC. On Instagram data, the precision of all RNN methods are similar and low; they have approximately the same performance as PVC.

The last message detector type uses the embedding representation shown in bottom right of Figs. 2 to 4. On Twitter, the embedding message learner by itself has the best precision. The embedding message learner when combined with node2vec user learner has the second-best precision on Twitter. On Ask.fm and Instagram, however, the combination of the embedding message learner and the node2vec user learner has the best precision.

Comparing across all models we considered, for all three datasets, the BoW message detectors, when combined with the node2vec user detector, had better precision. A similar trend occurs when using the doc2vec message learner. We believe the deep models, because they attempt to extract more semantic meaning of the words, are able to overcome the sparsity of our Instagram data. While the RNN message detector does better than PVC and the other baselines on Twitter, its performance is poor compared to PVC and baselines on Ask.fm and Instagram.

We summarize the precision analysis by answering three major questions:

(i) **Is there any combination that performs the best?** We list in Table I the precision@100 of all methods. We bold the top four methods with the highest precision@100 for each dataset. Then, we highlight the methods that are among the top four for all three datasets. The statistics indicate that the combinations of the node2vec user learner and the doc2vec or embedding message learners produce the most precise detections.

(ii) **Are deep models better than non-deep models?** We compute the average precision@100 score of deep models vs. non-deep models (PVC, seed-based, and naive-participant). For Twitter, the average precision@100 score of deep models and non-deep models are 0.541 and 0.286 respectively. For Ask.fm, the average score of deep models and non-deep models are 0.295 and 0.15 respectively. For Instagram, the average score of deep models is 0.1216, while the average score of non-deep models is 0.0766.

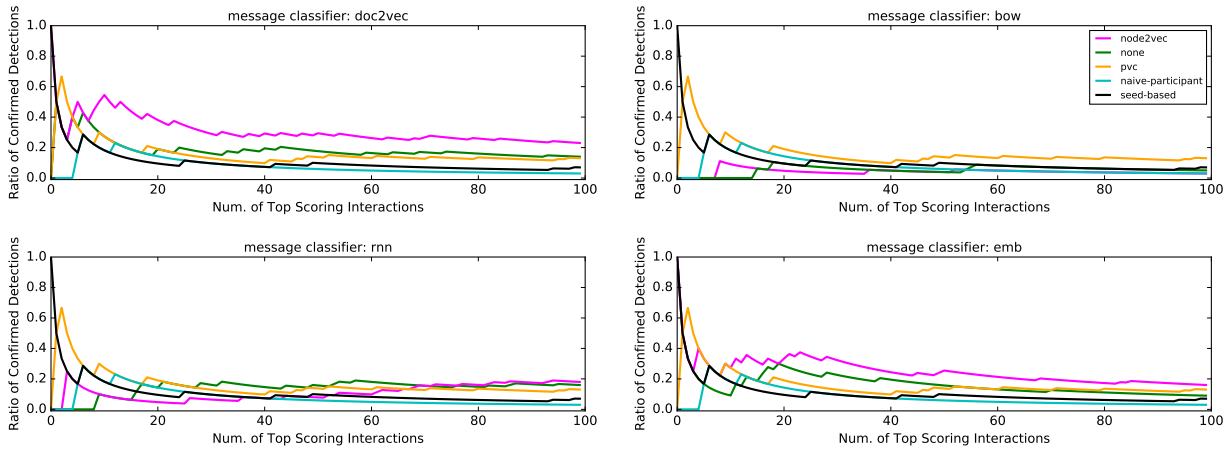

Fig. 4: Precision@k for bullying interactions on Instagram data using the combination of message and user learners, PVC, seed-based, and naive-participant.

In all three datasets, the average score of precision@100 for deep models is higher than the value in non-deep models. This trend suggests that on average, deep models outperform the non-deep models at detecting cyberbullying instances.

(iii) **Does co-training help?** Figure 5 plots the difference between the precision@100 score of the message learner co-trained with a user learner and the message learner by itself for each dataset. If this difference is positive, it indicates that the co-training helps improve precision. For Ask.fm, the co-training improves the precision for all datasets. For Instagram, the co-training improves the precision for all language models except the BoW learner. For Twitter, however, the co-training only helps the doc2vec message learner; the precision for the other two language models reduces slightly with co-training. These summary results suggest that co-training often provides significant improvements, but it can also somewhat reduce precision in some cases.

model	Ask.fm	Twitter	Instagram
doc2vec_node2vec	**0.43**	**0.6**	**0.23**
bow_node2vec	**0.4**	0.59	0.03
emb_node2vec	**0.33**	**0.65**	**0.16**
bow_none	**0.32**	**0.61**	0.05
doc2vec_none	0.31	0.55	0.14
rnn_node2vec	0.29	0.56	**0.18**
emb_none	0.26	**0.67**	0.09
rnn_none	0.25	**0.61**	**0.16**
pvc	0.28	0.32	0.13
seed-based	0.09	0.24	0.07
naive-participant	0.08	0.3	0.03

TABLE I: Precision@100 across all methods. The methods with the highest precision@100 for each dataset are bolded. The methods that are among the top four for all three datasets are highlighted. The doc2vec_node2vec and emb_node2vec combinations are among the top methods for all datasets.

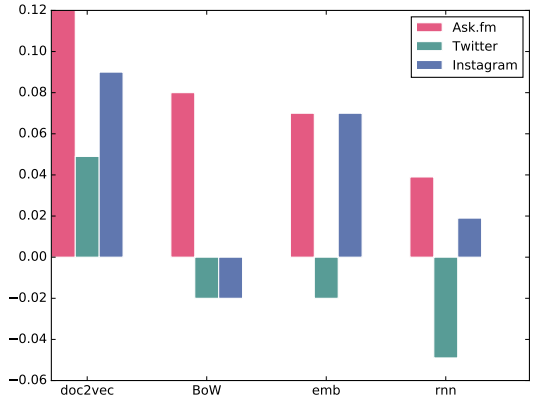

Fig. 5: The difference between the precision@100 score of the co-trained ensemble and the message learner trained alone. The improvements produced by co-training are more pronounced for the doc2vec language model than others, but the co-training also improves the performance of the other models as well.

B. *Qualitative Analysis*

We inspected the interactions the eight models identified in the three datasets and found three categories of note. The first category contains bullying conversations detected by most of the models and verified by the annotators. Examples in this category are straightforward true positives because most of these conversations have a high concentration of swear words. Two examples of such conversations follow.

User1:	youre not even sick you dumb b*tch
User2:	I cant f*cking breathe you ugly c*nt
User1:	then how is you alive dumb hoe stupid b*tch *ss c*nt. Kta b*tch

User1:	never seen another man on someones d*ck like you. Why you worried about him being ugly. Your prob a dogs *ss yourself.
User2:	just saw your avy lmaooooooooo youre just as ugly, you can take the cape off now freak
User1:	lol ok bud. Keep rtn shoes for sale to your 9 followers f*ckin loser.
User2:	look in the mirror before you can anyone a loser d*ck wad.

The second category of interactions contains conversations

with little prototypical bullying language, which are detected by models with the user learner but not by models without user classifiers (i.e., the BoW, doc2vec, RNN, and embedding message learners alone). Because the language-only detectors do not discover these types of conversations, these examples are evidence that considering social structure helps find complicated harassment incidents. Two examples of these challenging true positives follow.

User1:	Truth is. You hate me. Rate- my mom said if I have nothing nice to say, I shouldn't say anything at all.
User2:	Let me explain why I hate you. Okay so I only hate three people so obviously you have pissed me off enough to get on that list. So for starters, you obviously said you think that T*** and J*** will go to hell. Don't say two of best friends will go to hell because who else would T and J be? Second, you called R*** gay. That's not acceptable either. He even had a girlfriend at the time. You blamed it on your friend P**** or whatever her name is. So you didn't accept what you did and tried to hide it and that didn't work because we ALL know you called him gay multiple times. Another thing is, you are honestly so ignorant and arrogant. You think you are the best of the best and think you have the right to do whatever you want, whenever you want but you cant. I hate to break it to you, but you aren't the little princess you think you are. and you are basically calling me ugly in that rate. But you know what? i know im not the prettiest but at least im not the two-faced, conceited, b*tch who thinks that they can go around saying whatever they want. because saying people will go to hell can hurt more than you think. calling someone gay is really hurtful. youve called me ugly plenty of times, too. so congratulations you have made it on the list of people i hate. and i could go on and on but i think ill stop here. btw; your mom obviously didnt teach you that rule well enough. "buh-bye"

User1:	listen *—* you need to stop , leave me alone , stop harassing me . leave me alone your a creeper ....
User2:	Im harassing you ? baha . thats funny bc your the one that started with me , you were the one that said you were gonna fight me . and *—* is the one that has the videos . so get your facts right . and Im not gonna waste my time on you . why the hell would I do that ? baha .

The third category of interactions contain non-bullying conversations detected by most models. These false positives are considered by the annotators and us to be non-harassment. In many of these false-positive interactions, the users are joking with each other using offensive words, which is common among younger social media users. These examples include usage of offensive language but may require sophisticated natural language processing to differentiate from harassing usage. Two examples of these false positives follow.

User1:	Why you such a b*tch?
User2:	i have to stay sassy sl*t xx
User1:	Thanks.
User2:	youre f*cking welcome.

User1:	link plz you b*tch :P I Need A Better f*cking score :P :P
User2:	who you callin b*tch, b*tch :P
User1:	Motherf*cker :P
User2:	f*ckface. :P
User1:	Dipsh*t B*tch *sshole *ss :P

Many of the detections by our machine learning models appeared to be correct. Since most of the false positives that we observed were conversations with a high concentration of offensive language, we expect a more refined form of weak supervision than key phrases may help the co-training approach make more nuanced judgements about these cases. Nevertheless, our examination of the detected conversations provided evidence of how effective weakly supervised learning can be at training these deep learning models.

V. Conclusion

We present a method for detecting harassment-based cyberbullying using weak supervision. Harassment detection requires managing the time-varying nature of language, the difficulty of labeling the data, and the complexity of understanding the social structure behind these behaviors. We developed a weakly supervised framework in which two learners train each other to form a consensus on whether the social interaction is bullying by incorporating nonlinear embedding models.

The models are trained with an objective function consisting of two losses: a weak-supervision loss and a co-training loss that penalizes the inconsistency between the deep language-based learner and the deep user-based learner. We perform quantitative and qualitative evaluations on three social media datasets with a high concentration of harassment. Our experiments demonstrate that co-training of nonlinear models improves precision in most of the cases.

One of our future goals is to develop methods to train cyberbullying detection models to avoid learning discriminatory bias from the training data. A serious concern of any automated harassment or bullying detection is how differently they flag language used by or about particular groups of people. Our goal is to design fair models for cyberbullying analysis to prevent unintended discrimination against individuals based on sensitive characteristics including race, gender, religion, and sexual orientations. To tackle this phenomenon mathematically, we will add an unfairness penalty term to the co-trained ensemble framework. The basic idea is to penalize the model when we observe discrimination in the predictions. Another goal of ongoing work is to explore the usage of different user embedding models beyond node2vec. In experiments not shown here, we explored another user embedding strategy in which user vectors are directly trained during optimization of the framework. However, it was not effective perhaps because it did not incorporate any network structure information the way node2vec does. We will continue exploring other user-node embedding approaches that are being actively developed in the community. We also plan to extend the current framework in order to distinguish user roles as bullies or victims. In this extended framework, there would be three learners co-training each other; a message learner, a sender learner, and a receiver learner. Separating the user learner into sender and receiver learners would help identify the directional structure of bullying.

References

[1] American Academy of Child & Adolescent Psychiatry, "Facts for families guide. the American Academy of Child & Adolescent Psychiatry," *http://www.aacap.org/AACAP/*, 2016. [Online]. Available: http://www.aacap.org/AACAP/

[2] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1188–1196.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, 2013, pp. 3111–3119.

[5] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," *CoRR*, vol. abs/1607.00653, 2016. [Online]. Available: http://arxiv.org/abs/1607.00653

[6] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *CoRR*, vol. abs/1304.5634, 2013. [Online]. Available: http://arxiv.org/abs/1304.5634

[7] ——, "Multi-view learning with incomplete views," *IEEE Transactions on Image Processing*, vol. 24, pp. 5812–5825, 2015.

[8] E. A. Platanios, H. Poon, T. M. Mitchell, and E. Horvitz, "Estimating accuracy from unlabeled data: A probabilistic logic approach," *CoRR*, vol. abs/1705.07086, 2017. [Online]. Available: http://arxiv.org/abs/1705.07086

[9] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling, "Never-ending learning," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, ser. AAAI'15. AAAI Press, 2015, pp. 2302–2310. [Online]. Available: http://dl.acm.org/citation.cfm?id=2886521.2886641

[10] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, ser. COLT' 98. ACM, 1998, pp. 92–100.

[11] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Information Fusion*, vol. 38, pp. 43–54, 2017.

[12] M. Dadvar, F. de Jong, R. Ordelman, and D. Trieschnigg, "Improved cyberbullying detection using gender information," *Dutch-Belgian Information Retrieval Workshop*, pp. 23–25, February 2012.

[13] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "Detecting offensive language in social media to protect adolescent online safety," *Intl. Conf. on Social Computing*, pp. 71–80, 2012.

[14] K. Dinakar, R. Reichart, and H. Lieberman, "Modeling the detection of textual cyberbullying," *ICWSM Workshop on Social Mobile Web*, 2011.

[15] C. Van Hee, G. Jacobs, C. Emmery, B. Desmet, E. Lefever, B. Verhoeven, G. De Pauw, W. Daelemans, and V. Hoste, "Automatic Detection of Cyberbullying in Social Media Text," *ArXiv e-prints*, Jan. 2018.

[16] G. K. Pitsilis, H. Ramampiaro, and H. Langseth, "Detecting Offensive Language in Tweets Using Deep Learning," *ArXiv e-prints*, Jan. 2018.

[17] K. Reynolds, A. Kontostathis, and L. Edwards, "Using machine learning to detect cyberbullying," *Intl. Conf. on Machine Learning and Applications and Workshops (ICMLA)*, vol. 2, pp. 241–244, 2011.

[18] P. Bourgonje, J. M. Schneider, and G. Rehm, "Automatic classification of abusive language and personal attacks in various forms of online communication," in *Language Technologies for the Challenges of the Digital Age: Proceedings of the GSCL Conference 2017*, G. Rehm and T. Declerck, Eds. Springer, 2017.

[19] V. Nahar, X. Li, and C. Pang, "An effective approach for cyberbullying detection," *Communications in Information Science and Management Engineering*, vol. 3, no. 5, pp. 238–247, May 2013.

[20] D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, "Detection of harassment on Web 2.0," *Content Analysis in the WEB 2.0*, 2009.

[21] M. Ptaszynski, P. Dybala, T. Matsuba, F. Masui, R. Rzepka, and K. Araki, "Machine learning and affect analysis against cyber-bullying," in *Linguistic and Cognitive Approaches to Dialog Agents Symposium*, 2010, pp. 7–16.

[22] H. Margono, X. Yi, and G. K. Raikundalia, "Mining Indonesian cyber bullying patterns in social networks," *Proc. of the Australasian Computer Science Conference*, vol. 147, January 2014.

[23] Q. Huang and V. K. Singh, "Cyber bullying detection using social and textual analysis," *Proceedings of the International Workshop on Socially-Aware Multimedia*, pp. 3–6, 2014.

[24] N. Tahmasbi and E. Rastegari, "A socio-contextual approach in automated detection of cyberbullying," in *Proceedings of the 51stHawaii International Conference on System Sciences*, 2018, pp. 2151–2160.

[25] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Detecting aggressors and bullies on Twitter," in *Proceedings of the 26th International Conference on World Wide Web Companion*, ser. WWW '17 Companion, 2017, pp. 767–768. [Online]. Available: https://doi.org/10.1145/3041021.3054211

[26] V. K. Singh, Q. Huang, and P. K. Atrey, "Cyberbullying detection using probabilistic socio-textual information fusion," in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, vol. 00, Aug. 2016, pp. 884–887. [Online]. Available: doi.ieeecomputersociety.org/10.1109/ASONAM.2016.7752342

[27] D. Chatzakou, N. Kourtellis, J. Blackburn, E. D. Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on twitter," *CoRR*, vol. abs/1702.06877, 2017. [Online]. Available: http://arxiv.org/abs/1702.06877

[28] ——, "Measuring #gamergate: A tale of hate, sexism, and bullying," *CoRR*, vol. abs/1702.07784, 2017. [Online]. Available: http://arxiv.org/abs/1702.07784

[29] D. Chatzakou, N. Kourtellis, J. Blackburn, E. D. Cristofaro, G. Stringhini, and American Academy of Child & Adolescent Psychiatry, "Hate is not binary: Studying abusive behavior of #gamergate on twitter," *CoRR*, vol. abs/1705.03345, 2017. [Online]. Available: http://arxiv.org/abs/1705.03345

[30] C. Chelmis, D. Zois, and M. Yao, "Mining patterns of cyberbullying on twitter," in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, vol. 00, Nov. 2018, pp. 126–133. [Online]. Available: doi.ieeecomputersociety.org/10.1109/ICDMW.2017.22

[31] D.-S. Zois, A. Kapodistria, M. Yao, and C. Chelmis, "Optimal online cyberbullying detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE SigPort, 2018. [Online]. Available: http://sigport.org/2499

[32] T. Davidson, D. Warmsley, M. W. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," *CoRR*, vol. abs/1703.04009, 2017. [Online]. Available: http://arxiv.org/abs/1703.04009

[33] Z. Ashktorab and J. Vitak, "Designing cyberbullying mitigation and prevention solutions through participatory design with teenagers," in *Proc. of the CHI Conf. on Human Factors in Computing Systems*, 2016, pp. 3895–3905.

[34] M. H. Ribeiro, P. H. Calais, Y. A. Santos, V. A. F. Almeida, and W. M. Jr, ""like sheep among wolves": Characterizing hateful users on Twitter," *ArXiv e-prints*, Jan. 2018.

[35] H. Hosseinmardi, A. Ghasemianlangroodi, R. Han, Q. Lv, and S. Mishra, "Towards understanding cyberbullying behavior in a semi-anonymous social network," *IEEE/ACM International Conf. on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 244–252, August 2014.

[36] H. Hosseinmardi, S. Li, Z. Yang, Q. Lv, R. I. Rafiq, R. Han, and S. Mishra, "A comparison of common users across Instagram and Ask.fm to better understand cyberbullying," *IEEE Intl. Conf. on Big Data and Cloud Computing*, 2014.

[37] H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Analyzing labeled cyberbullying incidents on the Instagram social network," in *Intl. Conf. on Social Informatics*, 2015, pp. 49–66.

[38] ——, "Detection of cyberbullying incidents on the Instagram social network," *Association for the Advancement of Artificial Intelligence*, 2015.

[39] S. Tomkins, L. Getoor, Y. Chen, and Y. Zhang, "Detecting cyber-bullying from sparse data and inconsistent labels," in *NIPS Workshop on Learning with Limited Labeled Data*, 2017.

[40] E. Raisi and B. Huang, "Cyberbullying detection with weakly supervised machine learning," in *Proceedings of the IEEE/ACM International Conference on Social Networks Analysis and Mining*, 2017.

[41] ——, "Co-trained ensemble models for weakly supervised cyberbullying detection," in *NIPS Workshop on Learning with Limited Labeled Data*, 2017.

[42] K. Weinberger, A. Dasgupta, J. Langford, A. Smola, and J. Attenberg, "Feature hashing for large scale multitask learning," in *Proc. of the Intl. Conf. on Machine Learning*, 2009, pp. 1113–1120.

[43] noswearing.com, "List of swear words & curse words," *http://www.noswearing.com/dictionary*, 2016.

[44] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '04. New York, NY, USA: ACM, 2004, pp. 168–177. [Online]. Available: http://doi.acm.org/10.1145/1014052.1014073