A System to Monitor Cyberbullying based on Message Classification and Social Network Analysis

Stefano Menini[‡], Giovanni Moretti[‡], Michele Corazza[†]
Elena Cabrio[†], Sara Tonelli[‡], Serena Villata[†]
[‡]Fondazione Bruno Kessler, Trento, Italy
[†]Université Côte d'Azur, CNRS, Inria, I3S, France
{menini,moretti,satonelli}@fbk.eu
{michele.corazza}@inria.fr
{elena.cabrio,serena.villata}@unice.fr

Abstract

Social media platforms like Twitter and Instagram face a surge in cyberbullying phenomena against young users and need to develop scalable computational methods to limit the negative consequences of this kind of abuse. Despite the number of approaches recently proposed in the Natural Language Processing (NLP) research area for detecting different forms of abusive language, the issue of identifying cyberbullying phenomena at scale is still an unsolved problem. This is because of the need to couple abusive language detection on textual message with network analysis, so that repeated attacks against the same person can be identified. In this paper, we present a system to monitor cyberbullying phenomena by combining message classification and social network analysis. We evaluate the classification module on a data set built on Instagram messages, and we describe the cyberbullying monitoring user interface.

1 Introduction

The presence on social networks like Twitter, Facebook and Instagram is of main importance for teenagers, but this may also lead to undesirable and harmful situations. We refer to these forms of harassment as *cyberbullying*, i.e., 'an aggressive, intentional act carried out by a group or an individual, using electronic forms of contact, repeatedly and over time against a victim who cannot easily defend him or herself' (Smith et al., 2008). In online social media, each episode of online activity aimed at offending, menacing, harassing or stalking another person can be classified as a cyberbullying phenomenon. This is connected even with concrete public health issues, since recent studies show that victims are more likely to suffer from psycho-social difficulties and affective disorders (Tokunaga, 2010).

Given its societal impact, the implementation of cyberbullying detection systems, combining abusive language detection and social network analysis, has attracted a lot of attention in the last years (Tomkins et al., 2018; Hosseinmardi et al., 2015a; Ptaszynski et al., 2015; Dinakar et al., 2012). However, the adoption of such systems in real life is not straightforward and their use in a black box scenario is not desirable, given the negative effects misleading analyses could have on potential abusers and victims. A more transparent approach should be adopted, in which cyberbullying identification should be mediated by human judgment.

In this paper, we present a system for the monitoring of cyberbullying phenomena on social media. The system aims at supporting supervising persons (e.g., educators) at identifying potential cases of cyberbullying through an intuitive, easy-to-use interface. This displays both the outcome of a hate speech detection system and the network in which the messages are exchanged. Supervising persons can therefore monitor the escalation of hateful online exchanges and decide whether to intervene or not, similar to the workflow introduced in Michal et al. (2010). We evaluate the NLP classifier on a set of manually annotated data from Instagram, and detail the network extraction algorithm starting from 10 Manchester high schools. However, this is only one possible use case of the system, which can be employed over different kinds of data.

2 Network Extraction

Since cyberbullying is by definition a repeated attack towards a specific victim by one or more bullies, we include in the monitoring system an algorithm to identify local communities in social networks and isolate the messages exchanged only

Proceedings of the Third Workshop on Abusive Language Online, pages 105–110
Florence, Italy, August 1, 2019. ©2019 Association for Computational Linguistics

within such communities. In this demo, we focus on high-schools, but the approach can be extended to other communities of interest. Our case study concerns the network of Manchester high-school students, and we choose to focus on Instagram, since it is widely used by teenagers of that age.

Reconstructing local communities on Instagram is a challenging task. Indeed, differently from how other social networks operate (e.g., Facebook), Instagram does not provide a page for institutions such as High Schools, that therefore need to be inferred. To overcome this issue, and to identify local communities of students, we proceed in two steps that can be summarised as follow:

- *Expansion stage*. We start from few users that are very likely to be part of the local high school community, and we use them to identify an increasing number of other possible members expanding our network coverage.
- *Pruning stage*. We identify, within the large network, smaller communities of users and we isolate the ones composed by students. For these, we retrieve the exchanged messages in a given period of time (in our case, the ongoing school year), which will be used to identify abusive messages.

2.1 Expansion Stage

In this stage, we aim to build an inclusive network of people related to local high schools. Since schools do not have an Instagram account, we decide to exploit the geo-tagging of pictures. We manually define a list of 10 high schools from Manchester, and we search for all the photos associated with one of these locations by matching the geo-tagged addresses.

Given that anyone can tag a photo with the address of a school, this stage involves not only actual students, but also their teachers, parents, friends, alumni and so on. The reason to adopt this inclusive approach is that not every student is directly associated with his/her school on Instagram (i.e., by sharing pictures in or of the school), therefore we need to exploit also their contacts with other people directly related to the schools. We restrict our analysis to pictures taken from September 2018 on to focus on the current school year and obtain a network including actual students rather than alumni.

With this approach, we identify a first layer of 756 users, corresponding to the authors of the photos tagged in one of the 10 schools. Starting from these users, we expand our network with a broader second layer of users related to the first ones. We assume that users writing messages to each other are likely to be somehow related, therefore we include in the network all users exchanging comments with the first layer of users in the most recent posts. In this step, we do not consider the connections given by *likes*, since they are prone to introduce noise in the network. With this step we obtain a second layer of 17,810 users that we consider related to the previous ones as they interact with each other in the comments. Using the same strategy, we further expand the network with a third layer of users commenting the contents posted by users in the second layer. It is interesting to notice that in the first layer of users, i.e. the ones directly related to the schools, the groups of users associated with each school are well separated. As soon as we increase the size of the network with additional layers, user groups start to connect to each other through common “friends”.

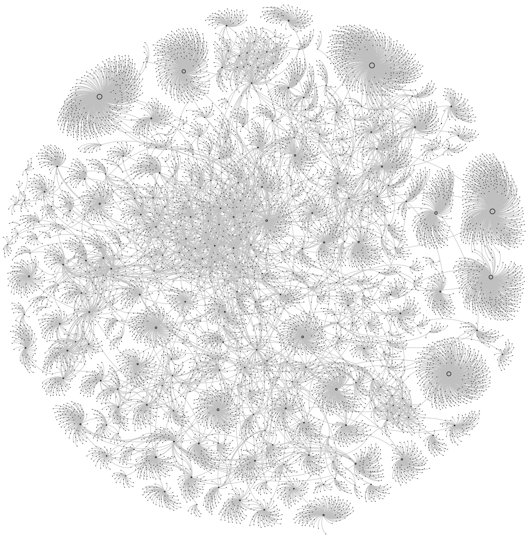


Figure 1: Network obtained starting from 10 Manchester schools and expanding +2 layers

We stop the expansion at a depth of three layers since additional layers would exponentially increase the number of users. At the end of the expansion stage, we gather a list of 544,371 unique users obtained from an exchange of 1,539,292 messages. The resulting network (Figure 1) is generated by representing each user as a node, while the exchanged messages correspond to edges. Each edge between two users is weighted according to the number of messages between the two.

2.2 Pruning Stage

After generating a large network of users starting from the list of schools, the following step consists in pruning the network from *unnecessary nodes* by identifying within the network *smaller communities* of high school students and teenagers. These communities define the scenario in which we want to monitor the possible presence of cyberbullying. To identify local communities, we proceed incrementally dividing the network into smaller portions. For this task, we apply the modularity function in Gephi (Blondel et al., 2008), a hierarchical decomposition algorithm that iteratively optimizes modularity for small communities, aggregating then nodes of the same community to build a new network.

Then, we remove the groups of people falling out of the scope of our investigation by automatically looking for geographical or professional cues in the user biographies. For example, we remove nodes that contain the term *blogger* or *photographer* in the bio, and all the nodes that are only connected to them in the network. This step is done automatically, but we manually check the nodes that have the highest centrality in the network before removing them, so as to ensure that we do not prune nodes of interest for our use case.

We then run again the modularity function to identify communities among the remaining nodes. Finally, we apply another pruning step by looking for other specific cues in the user bios that may identify our young demographic of interest. In this case, we define regular expressions to match the *age*, *year of birth* or *school attended*, reducing the network to a core of 892 nodes (users) and 2,435 edges, with a total of 14,565 messages (Figure 2).

3 Classification of abusive language

To classify the messages exchanged in the network extracted in the previous step as containing or not abusive language, we use a modular neural architecture for binary classification in Keras (Chollet et al., 2015), which uses a single feed-forward hidden layer of 100 neurons, with a ReLu activation and a single output with a sigmoid activation. The loss used to train the model is binary cross-entropy. We choose this particular architecture because it proved to be rather effective and robust: we used it to participate in two shared tasks for hate speech detection, one for Italian (Corazza et al., 2018a) and one for German (Corazza et al., 2018b), obtaining competitive results w.r.t. state-of-the-art systems.

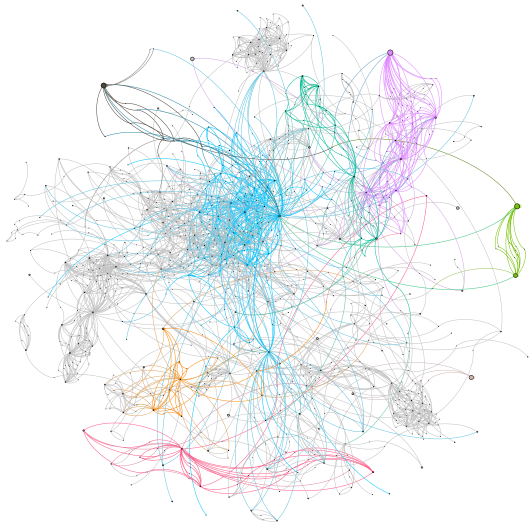


Figure 2: Manchester network after pruning

The architecture is built upon a recurrent layer, namely a Long Short-Term Memory (LSTM) whose goal is to learn an encoding derived from word embeddings, obtained as the output of the recurrent layer at the last timestep. We use English Fasttext embeddings[1] trained on Common Crawl with a size of 300. Concerning hyperparameters, our model uses no dropout and no batch normalization on the outputs of the hidden layer. Instead, a dropout on the recurrent units of the recurrent layers is used (Gal and Ghahramani, 2016) with value 0.2. We select a batch size of 32 for training and a size of 200 for the output (and hidden states) of the recurrent layers. Such hyperparameters and features have been selected from a system configuration that performed consistently well on the above mentioned shared tasks for hate speech detection, both on Facebook and on Twitter data.

4 Experimental setting and evaluation

Although our use case focuses on Instagram messages, we could not find available datasets from this social network with annotated comments. The widely used dataset used by (Hosseinmardi et al., 2015b) has indeed annotations at thread level.

We therefore train our classification algorithm using the dataset described in (Waseem and Hovy, 2016), containing 16k English tweets manually

[1] https://fasttext.cc/docs/en/english-vectors.html

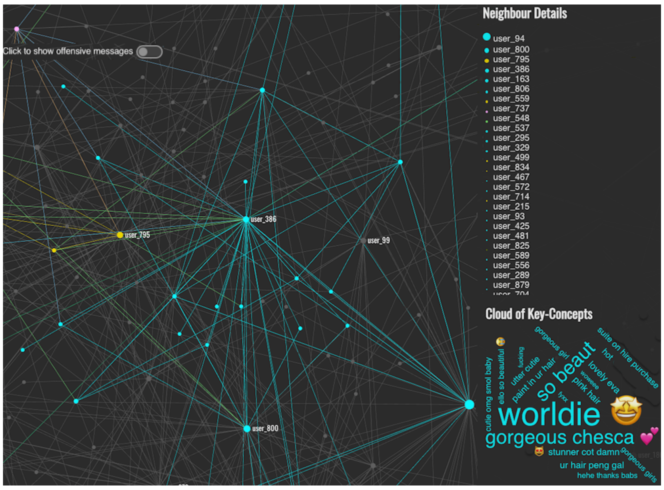
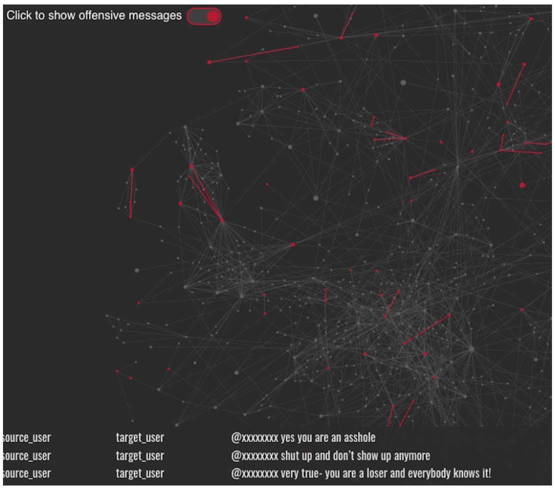


Figure 3: Interface view for network exploration



source_user target_user @xxxxxxxx yes you are an asshole
source_user target_user @xxxxxxxx shut up and don't show up anymore
source_user target_user @xxxxxxxx very true- you are a loser and everybody knows it!

Figure 4: Interface view for hate speech monitoring

annotated for hate speech. More precisely, 1,924 are annotated as containing racism, 3,082 as containing sexism, while 10,884 tweets are annotated as not containing offensive language. We merge the sexist and racist tweets in a single class, so that 5,006 tweets are considered as positive instances of hate speech. As a test set, we manually annotate 900 Instagram comments, randomly extracted from the Manchester network, labeling them as hate speech or not. Overall, the test set contains 787 non-offensive and 113 offensive messages.

We preprocess both data sets, given that hashtags, user mentions, links to external media and emojis are common in social media interactions. To normalize the text as much as possible while retaining all relevant semantic information, we first replace URLs with the word "url" and "@" user mentions with "username" by using regular expressions. We also use the Ekphrasis tool (Baziotis et al., 2017) to split hashtags into sequences of words, when possible.

The system obtained on the test set a micro-averaged F1 of 0.823. We then run the classifier on all messages extracted for the Manchester network, and make the output available through the platform interface.

5 Interface

The system[2] relies on a relational database and a tomcat application server. The interface is based on existing javascript libraries such as C3.js (`https://c3js.org`) and Sigma.js (`http://sigmajs.org`).

The platform can be used with two settings: in the first one (Figure 3), the Manchester network is displayed, with colors denoting different sub-communities characterised by dense connections. By clicking on a node, the platform displays the cloud of key-concepts automatically extracted from the conversations between the given user and her connections using the KD tool (Moretti et al., 2015). This view is useful to understand the size and the density of the network and to browse through the topics present in the threads. In the second setting (Figure 4), which can be activated by clicking on "Show offensive messages", the communities are all colored in grey, while the system highlights in red the messages classified as offensive by the system described in Section 3. By clicking on red edges it is possible to view the content of the messages classified as offensive, enabling also to check the quality of the classifier. This second view is meant to support educators and stakeholders in monitoring cyberbullying without focusing on single users, but rather keeping an eye on the whole network and zooming in only when hateful exchanges, flagged in red, are escalating.

6 Discussion

The current system has been designed to support the work of educators in schools, although it is not meant to be open to everyone but only to specific personnel. For example, in Italy there must be one responsible teacher to counter cyberbullying in every school, and access to the system could be given only to that specific person. For the same

[2] A video of the demo is available at `https://dh.fbk.eu/sites/dh.fbk.eu/files/creepdemo_1.m4v`

reason, the system does not show the actual usernames but only placeholders, and the possibility to de-anonymise the network of users could be activated only after cyberbullying phenomena have been identified, and only for the users involved in such cases. Indeed, we want to avoid the use of this kind of platforms for the continuous surveillance of students, and prevent a malicious use of the monitoring platform.

The system relies on public user profiles, and does not have access to content that users want to keep private. This limits the number of cyberbullying cases and hate messages in our use case, where detected abusive language concerns less than 1% of the messages, while a previous study on students' simulated WhatsApp chats around controversial topics reports that 41% of the collected tokens were offensive or abusive (Sprugnoli et al., 2018). This limitation is particularly relevant when dealing with Instagram, but the workflow presented in this paper can be potentially applied to other social networks and chat applications. Another limitation of working with Instagram is the fact that the monitoring cannot happen in real time. In fact, the steps to extract and prune the network require some processing time and cannot be performed on the fly, especially in case of large user networks. We estimate that the time needed to download the data, extract the network, retrieve and classify the messages and upload them in the visualisation tool would be around one week.

7 Conclusion

In this paper, we presented a platform to monitor cyberbullying phenomena that relies on two components: an algorithm to automatically detect online communities starting from geo-referenced online pictures, and a hate speech classifier. Both components have been combined in a single platform that, through two different views, allows educators to visualise the network of interest and to detect in which sub-communities hate speech is escalating. Although the evaluation has been carried out only on English, the system supports also Italian, and will be showcased in both languages. In the future, we plan to improve the classifier performance by extending the Twitter training set with more annotated data from Instagram. We will also experiment with cross-lingual strategies to train the classifier on English datasets and use it on other languages.

Acknowledgments

Part of this work was funded by the CREEP project (http://creep-project.eu/), a Digital Wellbeing Activity supported by EIT Digital in 2018 and 2019. This research was also supported by the HATEMETER project (http://hatemeter.eu/) within the EU Rights, Equality and Citizenship Programme 2014-2020.

References

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. DataStories at SemEval-2017 Task 4: Deep LSTM with Attention for Message-level and Topic-based Sentiment Analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

François Chollet et al. 2015. Keras. https://github.com/fchollet/keras.

Michele Corazza, Stefano Menini, Pinar Arslan, Rachele Sprugnoli, Elena Cabrio, Sara Tonelli, and Serena Villata. 2018a. Comparing Different Supervised Approaches to Hate Speech Detection. In *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, December 12-13, 2018.*, volume 2263 of *CEUR Workshop Proceedings*.

Michele Corazza, Stefano Menini, Pinar Arslan, Rachele Sprugnoli, Elena Cabrio, Sara Tonelli, and Serena Villata. 2018b. Inriafbk at germeval 2018: Identifying offensive tweets using recurrent neural networks. In *GermEval 2018 Workshop*.

Karthik Dinakar, Birago Jones, Catherine Havasi, Henry Lieberman, and Rosalind W. Picard. 2012. Common Sense Reasoning for Detection, Prevention, and Mitigation of Cyberbullying. *TiiS*, 2(3):18:1–18:30.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1019–1027.

Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard Han, Qin Lv, and Shivakant Mishra. 2015a. Analyzing labeled cyberbullying incidents on the instagram social network. In *Social Informatics - 7th International Conference, SocInfo 2015, Beijing, China, December 9-12, 2015, Proceedings*, pages 49–66.

Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard O. Han, Qin Lv, and Shivakant Mishra. 2015b. Prediction of Cyberbullying Incidents on the Instagram Social Network. *CoRR*, abs/1503.03909.

Ptaszynski Michal, Dybala Pawel, Matsuba Tatsuaki, Masui Fumito, Rzepka Rafal, Araki Kenji, and Momouchi Yoshio. 2010. In the service of online order: Tackling cyber-bullying with machine learning and affect analysis. *International Journal of Computational Linguistics Research*, 1(3):135–154.

Giovanni Moretti, Rachele Sprugnoli, and Sara Tonelli. 2015. Digging in the Dirt: Extracting Keyphrases from Texts with KD. In *Proceedings of the Second Italian Conference on Computational Linguistics*.

Michal Ptaszynski, Fumito Masui, Yasutomo Kimura, Rafal Rzepka, and Kenji Araki. 2015. Automatic extraction of harmful sentence patterns with application in cyberbullying detection. In *Human Language Technology. Challenges for Computer Science and Linguistics - 7th Language and Technology Conference, LTC 2015, Poznań, Poland, November 27-29, 2015, Revised Selected Papers*, pages 349–362.

Peter K. Smith, Jess Mahdavi, Manuel Carvalho, Sonja Fisher, Shanette Russell, and Neil Tippett. 2008. Cyberbullying: its nature and impact in secondary school pupils. *Journal of Child Psychology and Psychiatry*, 49(4):376–385.

Rachele Sprugnoli, Stefano Menini, Sara Tonelli, Filippo Oncini, and Enrico Piras. 2018. Creating a whatsapp dataset to study pre-teen cyberbullying. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 51–59. Association for Computational Linguistics.

Robert S. Tokunaga. 2010. Following you home from school: A critical review and synthesis of research on cyberbullying victimization. *Computers in Human Behavior*, 26(3):277 – 287.

Sabina Tomkins, Lise Getoor, Yunfei Chen, and Yi Zhang. 2018. A socio-linguistic model for cyberbullying detection. In *IEEE/ACM 2018 International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2018, Barcelona, Spain, August 28-31, 2018*, pages 53–60.

Zeerak Waseem and Dirk Hovy. 2016. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *SRW@HLT-NAACL*.