Hate speech prediction from Facebook Posts

Maliha Bushra Hoque
Dept. of CSE
BRAC University
Dhaka, Bangladesh
maliha.bushra.hoque@g.bracu.ac.bd

Sanjoy Dev
Dept. of CSE
BRAC University
Dhaka, Bangladesh
sanjoy.dev@g.bracu.ac.bd

Shishir Kumar Das
Dept. of CSE
BRAC University
Dhaka, Bangladesh
shishir.kumar.das@g.bracu.ac.bd

MD. Mustakin Alam
Dept. of CSE
BRAC University
Dhaka, Bangladesh
md.mustakin.alam@g.bracu.ac.bd

Md Humaion Kabir Mehedi
Dept. of CSE
BRAC University
Dhaka, Bangladesh
humaion.kabir.mehedi@g.bracu.ac.bd

Annajiat Alim Rasel
Dept. of CSE
BRAC University
Dhaka, Bangladesh
annajiat@gmail.com

***Abstract*—Social networking sites facilitate contact and information sharing, but they are often used to start damaging campaigns against particular people and organizations. Cyberbullying, encouragement of self-harm behaviors, and sexually predatory behavior are only a few of the serious consequences of widespread internet offensives. Additionally, attacks might target groups of victims and turn into physical violence. With our work, we want to stop and restrict the alarming spread of these hate campaigns. We take into account the linguistic content of comments made on a selection of publicly accessible Italian pages using Facebook as a benchmark. To categorize different types of hate, we first suggest a number of categories. According to the defined taxonomy, up to five different human annotators can subsequently annotate the crawled comments. We create and implement two classifiers for the Italian language using morpho-syntactical features, sentiment polarity, and word embedding lexicons. The first classifier is based on Support Vector Machines (SVM), and the second one is based on a specific Recurrent neural network called Long Short Term Memory (LSTM). We put these two learning algorithms to the test in order to confirm how well they classify hate speech.**

***Index Terms*—Social Networking, Cyberbullying, Violence, Vector Machines, embedding, morpho-syntactical features.**

I. Introduction

The best area for the users of Internet to stay in contact, discuss details their hobbies and everyday affairs, publish, and access documents, images, and videos is on social networking sites (SNSs). SNSs like Facebook, Twitter, Ask.fm, and Google+ enable users to build professional files, maintain a list of peers with whom they may connect, and post and read other users' posts. It is not surprising that search engines and SNSs are among the most popular websites overall.

Unfortunately, SNSs provide the perfect environment for the spread of false information. Cyberbullying, sexual predatory behavior [25], and exhortation to commit acts of self-harm [6] are only a few of the detrimental effects of the spread of bad information on SNSs. Although many of these attacks are frequently carried out by a single person, they can also be controlled by groups. Trolls frequently choose their victims, but there are times when hatred is focused towards large groups of people who are subject to prejudice due to characteristics like race or gender. Such campaigns may involve a sizable number of haters who become enthused by hostile discourse and who may act violently or physically as a result of their hatred.

The article [21] describes the attacker and defines trolls as internet users who seem to earnestly want to be a member of an online community but whose true aims are to cause trouble and stir up conflict for their own pleasure. So, sexists, religious fanatics, and political extremists frequently use SNSs to incite hatred toward particular people or organizations. Although more seasoned users might be able to deal with threats and trolls, the vast majority of them—especially children and those who might be mediatically subject to public criticism—cannot readily withstand the attacks. The media routinely presents proof of the (sadly dramatic in some cases) effects that gullible and emotional users have had to endure.

This project focuses on Italian literature and attempts to stop the worrisome spread of large-scale online hate campaigns. Various strategies have been used in the past to address the problem, which falls in between preemption and repair. One strategy seeks to reduce the amount of offensive content in chat chats with ad hoc filters, like in [38], by automatically identifying and eliminating it. The second method works with published content and attempts to delete the problematic one, frequently using analysis of numerous texts [3,7,36]. Contributions. Our goal is not to ban online information because we focus on classifying it for the Italian language in order to identify unusual waves of revulsion and hatred. We categorize the content of comments that appeared

on a selection of public pages using Facebook as a baseline. We provide support in the following areas :

We create the first hate speech classifier for Italian and contrast two methodologies based on cutting-edge learning algorithms for sentiment analysis tasks.

The findings reported in this research serve as a precursor to the identification of violent discussions as a whole, beginning with the classification of a single comment on a Facebook page. This is with the ultimate goal of quickly identifying hate waves in which multiple users may participate, as it tragically happened recently on Facebook pages.

The corpus for hate speech detection is introduced in the following section. Our classification methods are presented in Section 3 along with performance data. We go over similar research on identifying textual hostility on social media in Section 4. The paper is concluded in Section 5.

II. Literature Review or Related Works

Here, we review some scholarly articles on the identification of trolls and hate speech. It's interesting to note that links between user profiles on social networking sites are frequently directly tied to those between them in real life [16]. Machine learning has made it easy to identify users who utilize troll profiles for cyberbullying [4, 13, 18]. The contents of anonymous users on various opinion websites have also been linked together using text analysis techniques [1]. Lightweight profile elements were successful in identifying phony Twitter followers [11, 12], while relationships based on profile links and behaviors were successful in identifying false Facebook profiles [10]. [32] contains a seminal work on text classification for automatic hate speech detection. The authors of [27] provide a rule-based classifier to separate legitimate from offensive content in texts. PALADIN [24] is a pattern mining tool for mining linguistic patterns and identifying users' antisocial activities. Neural language models are used in the research in [15] to develop distributed low-dimensional representations of remarks. The method produces text embeddings that can be utilized as classifier input. The writers of [19] explain the difference between hate speech and flaming speech (the latter being more directed to groups, rather than individuals). The three-level hatred classification used in this paper is proposed in the same work (partially suffering for the low IAA too). Studies that focus on user behavior do so. The authors of [30] suggest a reputation tracking system that uses both positive and negative reviews to monitor user reputation. A behavioral analysis of banned users is in [7], which demonstrates some degree of textual similarity and frequently irrelevant content in their texts.

III. Dataset, Data description, and pre-processing

This section details the phase of our Italian hate speech corpus' retrieval and annotation.

1. Data Mining

We have created a corpus of comments taken from the public Facebook pages of Italian newspapers, politicians, artists, and groups in an effort to track the "hate level" on Facebook. These sections frequently hold debates on a wide range of subjects. We have created a flexible Facebook crawler that makes use of Graph API to retrieve comments on Facebook posts. The crawler makes use of the Laravel framework's adaptability to deploy a variety of capabilities, including adaptability, code reuse, various storage techniques, and parallel processing. It's been implemented as a Web service, and you can use a cURL command or a Web interface to control it. The tool needs certain target pages to crawl as well as a set of registered application keys. It has the ability to store data as JSON files, Kafka queues, or Elasticsearch indexes in the filesystem. It can crawl many pages simultaneously depending on how many application keys are given to the program. The crawler gathers all the data linked to postings, including comments on comments, starting with the most recent post. However, we have restricted our analysis in this work to direct comments on the postings in order to keep things simple.

2. Data Pre-processing

The crawler was used to gather comments about a number of websites and groups that we feared may contain hate speech.

Title of Facebook page	Annotated posts	Comments	Annotations
salviniofficial	19	5404	15298
matteorenziufficiale	2	158	584
lazanzarar24	10	307	1253
jenusdinazareth	2	132	460
sinistracazzateliberta2	7	79	234
ilfattoquotidiano	11	126	135
emosocazzi	4	73	75
noiconsalviniufficiale	14	223	270

Table 1: Dataset description and annotations

Overall, we gathered 17,567 Facebook comments from 99 posts that were crawled from the chosen pages: 6,502 of these had at least one annotation (spread across 66 posts), and at most 5 annotations from various human annotators. The majority of the comments received more than one annotation when we invited five bachelor students to do so. There were 5742, 3870, 4587, 2104, and 2006 comments from students, respectively. Particularly, 3,685 of the annotated comments

had three or more annotations. Each annotator added 3,662 comments on average. The classes, which range from "no hate" through "weak hate" to "strong hate," were given to the annotators to assign to each post. After that, we separated hate speech into many categories, including religion, physical or mental impairment, socioeconomic status, politics, race, sex and gender issues, and others.

IV. Methodology

On the annotated dataset, we compute a number of characteristics that are fully described in the sections that follow. A number of lexicons that were used to extract some of the attributes are detailed in Section 3.1.1. The classifier is then given the results of the annotation along with a vector of features that represent the comments in our dataset. The classifier gains the ability to categorize a comment based on the features' values and the results during the training phase. In the test phase, the classifier determines whether or not comments transmit hatred based on the trained model.

The Classifier

Two distinct classifiers were put to the test, one using Support Vector Machines (SVM) and the other using a specific Recurrent Neural Network called Long Short Term Memory (LSTM). Unfortunately, these kinds of methods capture "sparse" and "discrete" data in document classification tasks, despite the fact that SVM is a very powerful performer and cannot be easily surpassed. While this is frequently the most important aspect in determining the overall sentiment polarity of a document, it makes the detection of relations in sentences extremely difficult [33]. LSTM networks, on the other hand, are a subset of recurrent neural networks (RNN), which may identify long-term dependencies in a sentence. Because our method is dependent on morpho-syntactically marked texts, the hate speech corpus was mechanically morpho-syntactically tagged by the Part-Of-Speech tagger described in [14]. Both sentiment polarity and word embedding lexicons were applied to increase the overall accuracy of our system. Positive, negative, and neutral sentiment of Italian social media messages were correctly categorized using sentiment polarity lexicons [2]. We used two Italian corpora: I PAISA [26], a large corpus of authentic contemporary Italian texts; and a lemmatized corpus of 1,20 texts. In addition to these tools, we also developed two Word Embedding lexicons to address the problem that lexical details in brief texts are frequently lacking. We used word2vec to train two predict models for this purpose. These models pick up on word embeddings in lower dimensions. Each word is a multidimensional vector that represents a particular instantiation of the latent (hidden) variables used to represent embeddings. A tokenized version of the itWaC corpus was used to create the initial lexicon12. A tokenized corpus of tweets served as the foundation for the second lexicon. This corpus, which contains 8,400,382 Italian tweets, was compiled using the Twitter APIs.

The LSTM classifier

Hochreiter and Schmidhuber made the initial suggestion for the LSTM unit [22]. Potential long-distance dependencies can be captured by LSTM units because they can spread an important feature that appeared early in the input sequence over a large distance. Modern semantic composition performer LSTM computes the representation of a document from the representation of its words, with various degrees of abstraction. A low-dimensional, continuous, and real-valued vector is used to represent each word. We choose a bidirectional LSTM design because, by building bidirectional links in the network, it enables us to collect long-range dependencies from both directions of a document [31]. Additionally, we added a dropout component to the recurrent connections and input gates to avoid overfitting, a common problem with neural networks [17]. For this work, we have selected a dropout factor value in the ideal range [0.3, 0.5], more specifically 0.45, as proposed in [17]. Categorical cross-entropy is employed as a loss function in the optimization process, and the rmsprop optimizer was used to carry out the optimization [34].

The Support Vector Machine (SVM) classifier

Multiple linguistic description levels are covered by the wide range of features the Support Vector Machine classifier uses. The SVM algorithm's objective is to establish the best line or decision boundary that can divide n-dimensional space into classes, allowing us to quickly classify fresh data points in the future. A hyperplane is the name given to this optimal decision boundary.

V. Experimental Result Analysis:

Two different classification experiments were conducted:

Taking into account the three main types of hatred, the first (Strong hate, Weak hate, and No hate).

The second only considers two categories (Hate and No hate) with No hate being created by merging the Strong hate and Weak hate groups.

Only documents with at least three different annotations and those that belong to the most annotated class were used in the trials. Two datasets were produced as a result of this process: the two-class dataset, which consisted of 3,213 documents and was divided into 2,789 No hate and 786 Hate documents, and the three-class dataset, which

consisted of 2,416 documents and was divided into 2,116 No hate, 385 Weakhate, and 115 Strong hate documents. In order to balance the datasets, we chose a subset of the No hate texts that was only as large as the documents in the Weakhate class in the three-class trial and as large as the documents in the Hate class in the two-class experiment. 10-fold cross validation was used in this procedure to assess the performance of the two hate speech classifiers in the two studies. Ten separate, non-overlapping training and test sets were created from each dataset at random. The overall accuracy, F-score, Precision, and Recall for each class were calculated using the average of these values across all ten test sets. Evaluation measures including accuracy, precision, recall, and F-score are frequently employed in classification tasks. In our example: Recall expresses how many comments in the entire set have been correctly recognized.

The outcomes of the three-class experiment are reported in Table 2. The ability of SVM and LSTM to distinguish between the three classes is noteworthy, and the Strong hate one is a case in point. The low level of annotator agreement and the low amount of Strong hate papers (which is the class with the fewest documents) may be to blame for these outcomes. These findings prompted us to carry out the two-class experiment, the accuracy of which is shown in Table 3. The results are far better than those from the prior experiment, as we had anticipated. This is likely caused by both the higher annotator agreement for the three-class experiments and the increased amount of hate texts when compared to the Strong and Weak classes.

In our final experiment, we chose the documents for which at least 70 percent of the annotators were in agreement to assess the effects of annotator agreement on classification performance (321 Hate and 642 No-Hate documents). As Table 3 illustrates, the better the agreement, the more accurate both classification systems are. This improvement is especially significant for the Hate class classification, which has an F-score of almost 72 percent. These findings open the door for the application of our approach in a practical setting. The results also show that this Hate Speech corpus, which has been filtered in accordance with the annotator agreement, enables the creation of automatic hate speech classifiers with accuracy comparable to that attained in the majority of sentiment analysis tasks, like subjectivity and polarity classification. [2].

		Strong hate			**Weak hate**			**No hate**		
Classifier	Accuracy (%)	Prec.	Rec.	F-score	Prec.	Rec.	F-score	Prec.	Rec.	F-score
SVM	64.61	.452	.189	.256	.523	.525	.519	.724	.794	.757
LSTM	60.50	.501	.054	.097	.434	.159	.221	.618	.950	.747

Table 2: Ten-fold cross validation results on *Strong hate*, *Weak hate* and *No hate* classes.

		Hate			**No hate**		
Classifier	Accuracy (%)	Prec.	Rec.	F-score	Prec.	Rec.	F-score
SVM	72.95	.625	.568	.594	.778	.817	.797
LSTM	75.23	.640	.6832	.657	.824	.791	.805
		$\geq$ 70% of Agreement					
SVM	80.60	.757	.689	.718	.833	.872	.851
LSTM	79.81	.706	.758	.728	.859	.822	.838

Table 3: Ten-fold cross validation results on *Hate* and *No hate* classes.

VI. Conclusion

This paper developed an Italian text classification system for hate speech. When using a binary classification, the classifier produced results for Italian that were equivalent to those of the majority of evaluated sentiment analysis tasks. The output from the two classifiers will be thoroughly analyzed to see whether their performance can be improved by combining them. Additionally, we are expanding the annotation process to gather more annotations for a single comment and to expand the corpus size. Additionally, we'll look into how sarcasm affects classifier performance. We think this approach serves as the foundation for tracking diverging states of Italian texts in online dialogues because human moderators cannot keep an eye on the massive user-generated texts on social networks.

References

[1] Mishari Almishari and Gene Tsudik. Exploring linkability of user reviews. In ESORICS, pages 307–324. Springer Berlin Heidelberg, 2012.

[2] Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. Overview of the Evalita 2014 sentiment polarity classification task. In EVALITA, 2014.

[3] Peter Burnap and Matthew Leighton Williams. Hate speech, machine classification and statistical modelling of information flows on Twitter. In Internet, Policy and Politics, 2014.

[4] Erik Cambria et al. Do not feel the trolls. In Semantic Web, 2010.

[5] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1–27:27, 2011.

[6] S. Chattopadhyay et al. Suicidal risk evaluation using a similarity-based classifier. In Advanced Data Mining and Applications, pages 51–61. Springer Berlin Heidelberg, 2008.

[7] Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. Antisocial behavior in online discussion communities. arXiv preprint arXiv:1504.00680, 2015.

[8] Fran¸cois Chollet. Keras. https://github.com/fchollet/keras, 2015.

[9] Andrea Cimino, Stefano Cresci, Felice Dell'Orletta, and Maurizio. Tesconi. Linguisticallymotivated and lexicon features for sentiment analysis of italian tweets. In EVALITA, 2014.

[10] Mauro Conti, Radha Poovendran, and Marco Secchiero. FakeBook: Detecting fake profiles in on-line social networks. In Social Networks Analysis and Mining, pages 1071–1078. IEEE, 2012.

[11] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. A criticism to society (as seen by twitter analytics). In ICDCS Workshops, pages 194–200, 2014.

[12] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. Fame for sale: Efficient detection of fake twitter followers. Decision Support Sys., 80:56–71, 2015.

[13] Maral Dadvar et al. Improving cyberbullying detection with user context. In Advances in Information Retrieval, pages 693–696. Springer, 2013.

[14] Felice Dell'Orletta. Ensemble system for part-of-speech tagging. Proceedings of EVALITA, 2009. 94

[15] Nemanja Djuric et al. Hate speech detection with comment embeddings. In 24th International Conference on World Wide Web, pages 29–30. ACM, 2015.

[16] Nicole B. Ellison and Danah M. Boyd. Sociality through social network sites. In The Oxford Handbook of Internet Studies. Oxford Handbooks Online, 2013.

[17] Yarin Gal. A theoretically grounded application of dropout in recurrent neural networks. arXiv preprint arXiv:1512.05287, 2015.

[18] Patxi Gal´an-Garc´ıa et al. Supervised machine learning for the detection of troll profiles in Twitter social network. In Joint Conf. Soco-Cisis-Iceute, pages 419–428. Springer, 2014.

[19] Njagi Dennis Gitari et al. A lexicon-based approach for hate speech detection. Multimedia and Ubiquitous Engineering, 10(4):215–230, 2015.

[20] Kilem L Gwet. Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters. Advanced Analytics, LLC, 2014.

[21] Claire Hardaker. Trolling in asynchronous computer-mediated communication: From user discussions to academic definitions. Politeness Research, 6(2):215–242, 2010.

[22] Sepp Hochreiter and J¨urgen Schmidhuber. Long short-term memory. Neural computation , 9(8):1735–1780, 1997.

[23] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In Knowledge discovery and data mining, pages 168–177. ACM, 2004.

[24] Ralf Klamma, Marc Spaniol, and Dimitar Denev. Paladin: A pattern based approach to knowledge discovery in digital social networks. In I-KNOW, volume 6, pages 6–8, 2006.

[25] April Kontostathis. Chatcoder: Toward the tracking and categorization of internet predators. In Text Mining Workshop of Siam Data Mining (SDM), 2009.

[26] Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. The PAISA corpus of Italian web texts. In Web as Corpus Workshop (WaC-9), 2014.

[27] Altaf Mahmud, Kazi Zubair Ahmed, and Mumit Khan. Detecting flames and insults in text. In Natural Language Processing, 2008.

[28] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.

[29] Preslav Nakov et al. Semeval-2016 task 4: Sentiment analysis in Twitter. In SemEval@NAACLHLT 2016, pages 1–18, 2016.

[30] F Javier Ortega. Detection of dishonest behaviors in on-line networks using graph-based ranking techniques. AI Communications, 26(3):327–329, 2013.

[31] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 45(11):2673–2681, 1997.

[32] Ellen Spertus. Smokey: Automatic recognition of hostile messages. In AAAI/IAAI, 1997.

[33] Duyu Tang et al. Document modeling with gated recurrent neural network for sentiment classification. In Empirical Methods in Natural Language Processing, pages 1422–1432, 2015.

[34] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[35] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phraselevel sentiment analysis. In HLT/EMNLP, pages 347–354. ACL, 2005.

[36] Guang Xiang et al. Detecting offensive tweets via topical feature discovery over a large scale Twitter corpus. In Information and Knowledge Management, pages 1980–1984. ACM, 2012.

[37] XingYi Xu et al. UNIMELB at SemEval-2016 tasks 4a and 4b: An ensemble of neural networks and a Word2Vec based model for sentiment classification. In SemEval, 2016.

[38] Zhi Xu and Sencun Zhu. Filtering offensive language in online communities using grammatical relations. In Collaboration, Electronic Messaging, Anti-Abuse and Spam, pages 1–10, 2010.