Hate Speech in Philippine Election-Related Tweets: Automatic Detection and Classification Using Natural Language Processing

Neil Vicente P. Cabasag[1], Vicente Raphael C. Chan[1], Sean Christian Y. Lim[1], Mark Edward M. Gonzales[1], Charibeth K. Cheng[2]

[1]De La Salle University Integrated School–Manila, [2]De La Salle University, College of Computer Studies

{neil_vicente_cabasag, vicente_raphael_chan, sean_christian_lim, mark_gonzales, charibeth.cheng}@dlsu.edu.ph

ABSTRACT

Social networking sites have opened avenues for the expression of disparaging and antagonistic sentiments, proliferating hate speech. While technologies have been devised to address this problem, systems contextualized in the Philippine cyberspace are essential since hate speech is deeply tied to the context of a locale. This research sought to address this need by developing a model capable of automating hate speech detection. Tweets posted during the 2016 Philippine electoral campaign was labeled as either hate- or non-hate-containing and annotated with the target(s) of hate. Simple language-independent features, namely, term frequency–inverse document frequency (TF-IDF), term occurrence (TO), and their combination, were extracted. For binary classification, logistic regression using TF-IDF+TO and with hashtag segmentation performed best (F1 = 77.47%), outperforming the keyword-matching rule-based classifier by around 6%. The feedforward neural network failed to outperform the best logistic regression model entirely but scored competitively and used fewer features. For multilabel classification, perceptron using TF-IDF+TO and with hashtag segmentation performed best (micro-F1 = 67.80%, macro-F1 = 61.86%), outperforming the rule-based classifier by 15.71% and 7.25% macro- and micro-F1, respectively. The main contribution of this paper is a comparative investigation of different classifiers using simple language-independent features for detecting and classifying political hate speech from the Philippines.

Keywords

Hate speech, text classification, machine learning, feedforward neural network, Twitter, social media, election

1. INTRODUCTION

In the present digital era, social networking sites have served as ubiquitous platforms that facilitate the widespread and rapid diffusion of diverse user-generated content. The advantages afforded by these avenues make it easy for people to engage not only in discourse but also in the expression of negative, offensive, and even discriminatory sentiments with minimal restriction of censorship. The emphasis of these media on speed, accessibility, and anonymity compounds this problem in relation to hate speech, which has been defined as:

> [A] bias-motivated, hostile, malicious speech aimed at a person or a group of people because of some of their actual or perceived innate characteristics. It expresses discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial attitudes towards those characteristics, which include gender, race, religion, ethnicity, color, national origin, disability, or sexual orientation. [1]

Although already evident in its myriad of forms throughout several episodes in history, its exacerbated proliferation at present can be ascribed to an interplay of both psychological proclivities and technological channels that feed these motivations. One particular concept to explain this phenomenon is the notion of filter bubbles and echo chambers [2], which breed communities of like-minded people to the detrimental exclusion of other viewpoints and opinions [3]. These exclusive digital communities are made more pronounced by computational algorithms with regard to narrative ranking, personalization settings, page recommendations, as well as group suggestions [4, 5].

It is incorrect, however, to assume that abusive language exists solely in the digital realm. Since digital interaction allows for a greater degree of anonymity and a wider network of connections than in face-to-face communication, the fundamental psychological proclivities that lead to the formation of filter bubbles are noticeably more amplified, heightening stereotyping, polarization, and radicalization [6]. The rise in the number of hate crimes can be attributed to these same factors [7, 8].

As a matter of fact, this is corroborated by Müller and Schwarz [9]; employing back-of-the-envelope regression calculations, they found an association between the absence of anti-refugee posts on the Facebook page of the right-wing party Alternative für Deutschland and a 9% decrease in real-world anti-refugee incidents.

Given the highly-polarizing nature of political discourse, its close relationship with the hate speech phenomenon becomes clear. Focus group participants from Kosovo remarked that denigrating rhetoric is most commonly employed by politicians in their attempt to demean the opposition, marginalize certain societal segments, willingly incite divisive sentiments, and divert the citizens' attention from pressing issues [10].

Campaign periods especially serve as hotbeds for inflammatory language directed towards candidates, parties, and sectors. During the 2015 Nigerian general election, denigrating advertisements and tirades, fomenting acts of violence, persisted between the major political parties in the young democracy [11, 12]. Even established democracies struggle with this problem as is the case with the 2016 United States presidential elections, with an uptick in misogynistic and sexist language beleaguering Hillary Clinton, the first female presidential candidate in the country [13]. These findings and cases are indicative of the power of hate speech as a political weapon that "neither promotes majoritarian democracy nor protects minority rights" [14].

In the Philippines, however, systematic quantitative and qualitative investigations on this matter have been exiguous. While the country is a signatory in international treaties, such as the 1965 International

Philippine Computing Journal Dedicated Issue on Natural Language Processing, pages 1–14
Vol. XIV No. 1 August 2019

Convention on Elimination on All Forms of Racial Discrimination and 1976 International Convention on Civil and Political Rights, there is no existing legal mechanism that explicitly and specifically penalizes hate speech [15]. The closest provisions in relation to contemptuous language would be the libel and oral defamation articles in the Revised Penal Code [16]. Nevertheless, a proposed Magna Carta for Philippine Internet Freedom (which includes a section on hate speech) and Hate Speech Act have been introduced in the upper and lower chambers, respectively [15, 17].

These considerations all point out to the undesirability of continued hate speech proliferation, which makes it imperative to enact counteractions; a possible solution would be the development of technological interventions. Hate speech detection is an active research topic in the domain of natural language processing (NLP), a subfield of artificial intelligence that is concerned with enabling computers to understand and analyze human language. Existing technologies have explored the building of classifiers based on *classic feature extraction* [18, 19, 20], which hinges on manual feature engineering for use of the classifier, and *deep learning algorithms* [21, 22], which attempt to mimic the human brain through the use of multiple stacked layers.

Research endeavors in the Philippines, however, have been limited. Most of the systems were developed by foreign researchers; their dataset, as well as the underlying frameworks and assumptions, may not be reflective of the culture and context of hate speech in the country's cyberspace.

This study seeks to contribute to the filling of this gap through the development of a model that can automate hate speech detection and classification in Philippine election-related tweets. The role of the microblogging site Twitter as a platform for the expression of support and hate during the 2016 Philippine presidential election has been supported in news reports and systematic studies [23, 24]. Thus, the particular question addressed in this paper is: Can existing techniques in language processing and machine learning be applied to detect hate speech in the Philippine election context?

Specifically, this research aims to:

- Review existing NLP methods employed in hate speech detection and classification, alongside techniques that can be utilized to extract features from hate-containing tweets;
- Analyze the targets of hate as seen in tweets posted during the campaign period for the 2016 Philippine presidential election;
- Implement hate speech detection and classification models following (1) rule-based, (2) machine learning, and (3) deep learning approaches; and
- Evaluate and compare the performance of the built models.

2. RELATED WORKS

The earliest work on the detection of hate speech, then defined as *abusive messages* or *flames*, appeared in 1997 with the prototype system Smokey developed by Spertus [25]. Employing a decision tree generator to identify linguistic rules associated with flaming, the system managed to correctly classify 64% of the flames and 98% of the non-flames. Among the features considered were the use of second-person pronouns to begin a sentence, noun phrases as appositions to second-person pronouns, imperative commands, and tag questions implying condescension.

Succeeding works have further explored the use of both classical feature extraction and deep learning [26]. While the former requires manual feature engineering, i.e., choosing features and translating them into vectors that will be used by the classifier, the latter takes inspiration from the neural connections in the biological brain and utilizes neural networks to automate feature learning. Schmidt and Wiegand [27] enumerated and surveyed the following key features that can be extracted for use in hate speech detection: simple surface features, word generalizations, sentiment analysis, lexical resources, linguistic features, knowledge-based features, meta-information, and multimodal information.

Simple surface features include bag of words and *n*-grams; notably, because hate speech often contains non-canonical spellings marked by omissions or amalgamations of symbols and alphanumeric characters in a single string (e.g., "t@ng1n@ m0 gagu," which gives "tangina mo gago" upon spelling correction), character-level approaches may be necessary to capture similarity to the canonical spelling. This is supported by the findings of Mehdad and Tetrault [28]: recurrent neural network language model and support vector machine with naïve Bayes features using token *n*-grams were bested by their character-level counterparts by seven and three F1-points, respectively. The effectiveness of character *n*-grams was also supported in a study conducted by Nobata *et al.* [29].

Although simple surface features have the advantage of being computationally inexpensive, classifiers based solely on these are restricted to capturing only crude textual features. One way to address this is through the introduction of w*ord generalization*, which can be done via methods such as topic modeling and embeddings. Latent Dirichlet allocation (LDA) was used by Xiang *et al.* [30] to generate topical features, yielding a 5.4%-increase in the number of true positive detections as compared to keyword matching. Different variations of embeddings, including word [31] and paragraph [32], have been experimented on as well.

The affect and polarity of a statement are the foci of *sentiment analysis*. This proves to be a viable approach since hateful content can be taken as a form of intense negative sentiment although it is necessary to avoid conflating the two, thus leading to multi-step classification approaches as employed by Sood, Churchill, and Antin [33]. The use of *lexical resources* or compiled dictionaries of keywords that signal different forms of hate, such as the *Insulting and Abusing Language Dictionary* [34] and the dictionary of hate verbs built by Gitari *et al.* [35] and seeded from an initial list of six verbs (*discriminate, loot, riot, beat, kill, and evict*), can also help in building classifiers based on sentiment analysis or keyword matching.

Linguistic features have also been employed as an augmentation to surface-level features as is the case with the study conducted by Xu *et al.* [36], which considered three sets of feature representations (unigrams, unigrams and bigrams, and part-of-speech-colored unigrams and bigrams) and four classifiers (naïve Bayes, linear- and radial basis function-kernelled support vector machine, and maximum entropy-equivalent logistic regression). However, it was found that part-of-speech tagging failed to significantly contribute to increase in performance. Additionally, typed dependencies have been integrated to infer relationships between words and possibly glean the sociological "us vs. them" divide associated with hate speech [37, 38].

The last three features mentioned by Schmidt and Wiegand [27] account for elements beyond content, i.e., beyond the given textual data. However, due to the expensiveness and difficulty of taking context into account, there are markedly fewer related studies. An example of a *knowledge base* is BullySpace [39]. Anchored on the extant semantic network ConceptNet and matrix AnalogySpace, it

is capable of detecting insults directed towards the lesbian, gay, bisexual, and transsexual community which, outside any presumed context, may be read as entirely innocuous. A significant limitation of such knowledge bases is their specificity to particular subgroups.

Social networking platforms have also made it possible for *meta-information*, or data about data, to be easily retrieved and analyzed for its predictive power although this comes with the hurdle of possibly crawling inauthentic personal information supplied by a user. Dadvar *et al.* [40] found that supplementing content-based with cyberbullying-indicative and user-based features (activity history and age of the user) yielded better precision, recall, and F1-scores than when user-based features are omitted. Waseem and Hovy [41] investigated the addition of demographic information and reported that the combination of 2- to 4-grams and gender registered the highest F1 at 73.89%.

A *multimodal information* approach capitalizes on the intersection of text, audio, image, and video dimensions in posts circulating on social media. The research of Hosseinmardi *et al.* [42], which focused on Instagram, showed that images and metadata contribute to cyberbullying detection as evinced in the built maximum entropy classifier registering recall and precision scores of 76% and 62%, respectively. A similar hybrid approach, coupled with the usage of LDA for caption topic generation and a pre-trained convolutional neural network (CNN) for image processing, was also employed by Zhong *et al.* [43].

For *deep learning,* Zhang and Luo [26] noted that the most widely-used architectures in researches are recurrent neural network, usually long short-term memory (LSTM) and CNN. Zimmerman, Fox, and Kruschwitz [44] combined CNN and word embeddings, outperforming the original best method by Waseem and Hovy [41]. Introducing a novel hybrid approach, Zhang, Robinson, and Tepper [45] designed a CNN+GRU (gated recurrent unit) architecture, which had the advantage of having a faster training time and yielding better generalizability compared to the usual LSTM. Feedforward neural network (deep multilayer perceptron) has also been employed. Polignano and Basile [46] integrated this as part of an ensemble in HanSEL, an Italian hate speech detection system, and achieved cross-validation F1 scores of 80.34% and 71.02% for Facebook sentences and Twitter posts, respectively.

Despite these existing technologies, hate speech detection remains a challenge. The definition of *hate speech* itself is problematic [47]; while a lax set of criteria may lead to several unidentified instances, a scrupulous one may come into conflict with the legally-enshrined freedom of expressing dissenting opinions. The annotation process itself is a difficult task, as evidenced by Ross *et al.* [48] reporting that even having a set definition failed to substantially increase reliability, which was already very low (with Krippendorff's alpha ranging from 0.18 to 0.29).

Context and domain-specific knowledge also play an important role in determining whether a given text is hateful or not, especially when vituperative intent is masked using sarcasm, humor, and code words. Furthermore, the Filipino language poses an interesting case in relation to textual analysis in general and hate speech detection in particular. Besides its speakers' knowledge of English and one or more local languages [49] leading to frequent code-switching, Filipino is also characterized by a rich and complex morphology that allows for a dynamically-expanding vocabulary [50]. These aforementioned challenges present several NLP opportunities for the improvement of hate speech recognition in online spaces, with special attention given on national and local languages and dialects.

3. METHODOLOGY

The research process consisted of the following phases: data collection, tweet labeling, data preprocessing, feature extraction, classifier building, and performance evaluation. An overview of the methodology is shown in Figure 1.

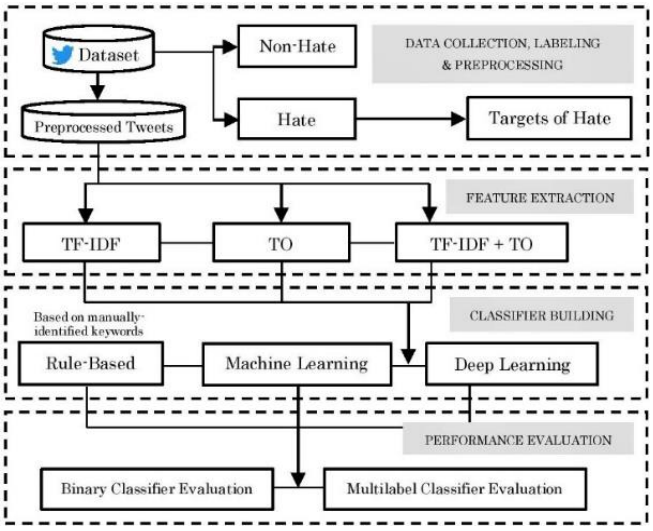

Figure 1. Overview of the Research Process

3.1 Data Collection

The dataset used in this study was a subset of the corpus 1,696,613 tweets crawled by Andrade *et al.* [24] and posted from November 2015 to May 2016 during the campaign period for the Philippine presidential election. They were culled based on the presence of candidate names (e.g., Binay, Duterte, Poe, Roxas, and Santiago) and election-related hashtags (e.g., #Halalan2016, #Eleksyon2016, and #PiliPinas2016).

3.2 Tweet Labeling

Two levels of data annotation were done: (1) labeling the tweets as either hate- or non-hate-containing and (2) labeling hate-containing tweets with the target(s) of hate. Hence, the first level is binary whereas the second is multilabel.

For the first level of annotation, the general definition used was the one given by Cohen-Almagor [1] and quoted in the introduction of this paper. To promote inter-annotator reliability, this definition was expounded through a set of guidelines, partially derived from the definition of an *offensive tweet* by Waseem and Hovy [41].

In particular, a tweet was considered *hate-containing* if it met at least one of the criteria enumerated below:

- It contained a profane word, a slur, or an epithet used in a discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial manner.
- Its meaning or intent was ambiguous due to lack of context or sparsity of words, but the tweet contained a profane word, a slur, or an epithet.
- It expressed dissent or criticism directed towards a group or individual in a discriminatory, intimidating, disapproving, antagonistic manner. This criterion holds regardless of the veracity or the plausibility of the claim, and/or the presence of an accompanying argument to support or explain it.

- It contained a stereotype, generalization, or characterization of a group or individual in a discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial manner.
- It defended a discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial stereotype, generalization, or characterization.
- It sought to silence a group or individual.
- It promoted hate speech or violent crimes.
- It contained a screen name or shows support for a hashtag that is deemed discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial.

Additional guidelines for labeling are as follows:

- Tweets that satisfied at least one of the criteria presented above were considered *hate-containing*, even if the delivery or expression is masked using humor or sarcasm.
- News briefs and news report excerpts tweeted by media outlets were considered *non-hate-containing*. However, tweets that included a comment or reaction to those news briefs or excerpts were evaluated based on the set of criteria presented above.
- Tweets that may be read ambiguously without the proper context but do not contain direct statements of hate were considered *non-hate-containing*.

These set of criteria differ from the one proposed by Waseem and Hovy [41] in three salient aspects. First, while Waseem and Hovy [41] deemed a tweet offensive if it "uses a sexist or racial slur," the present researchers added the condition that the slur or epithet should be used in the manner described by Cohen-Almagor [1]: "discriminatory, intimidating, disapproving, antagonistic, and/or prejudicial." Warner and Hirschberg [51] mentioned the following example of a non-hateful sentence containing a racial slur: *Kike is a word often used when trying to offend a jew.*

Second, while Waseem and Hovy [41] placed an emphasis on minorities as the targets of offensive tweets, the present researchers, following the definition by Cohen-Almagor [1], expressly widened the targets to include individuals and groups in general since the focus of this research is on political hate speech; as such, a number of the hate-containing tweets target specific candidates, parties, or groups that are not necessarily members of a minority segment.

Lastly, while the criteria of Waseem and Hovy [41] stipulated that a tweet criticizing a minority is offensive if it "uses a straw man argument" or if it is done "without a well-founded argument," the present researches considered all tweets that expressed dissent or criticism in the manner described by Cohen-Almagor [1] as hateful. Taking into account the high degree of polarization pervading the electoral campaign period, evaluating the substance of an argument or claim presented in the tweet may introduce biases on the part of the annotators, thus affecting the quality of the labels.

For the second level of annotation, those that were labeled as hate were tagged with the target(s). The typology, which was adapted from the classification scheme by Silva *et al.* [52], consisted of the following categories: *race, sex, physical, disability, religion, class*, and *quality*. Their definitions are presented in Table 1.

Table 1. Targets of Hate

Target	Definition
Race	Expresses hate towards or on the basis of race, ethnicity, or nationality; or associates an individual or a group to such in a hateful manner
Physical	Expresses hate towards or on the basis of a physical characteristic; or associates an individual or a group to such in a hateful manner
Sex	Expresses hate towards or on the basis of gender or sexual orientation; associates an individual or a group to such in a hateful manner; or includes a hateful remark or threat of a sexual nature
Disability	Expresses hate towards or on the basis of a health condition (including but not limited to a physical, mental, sensory, or emotional disability or impairment); or associates an individual or a group to such in a hateful manner
Religion	Expresses hate towards or on the basis of religious affiliation or belief; or associates an individual or a group to such in a hateful manner
Class	Expresses hate towards or on the basis of social class or socioeconomic status; or associates an individual or a group to such in a hateful manner
Quality	Expresses hate towards or on the basis of a quality that does not fall under any of the previously-mentioned targets; or associates an individual or a group to such in a hateful manner

The gold standard contained 729 tweets annotated independently by four of the researchers. Their agreement, which is reported in Table 2, was measured using Fleiss' kappa (κ) and prevalence adjusted bias-adjusted kappa (PABAK):

$$\text{Fleiss' } \kappa = \frac{\text{observed agreement} - \text{chance agreement}}{1 - \text{chance agreement}} \quad \text{(Equation 1)}$$

$$\text{PABAK} = 2 \cdot \text{observed agreement} - 1 \quad \text{(Equation 2)}$$

Table 2. Inter-annotator Agreement

Label	Fleiss' κ	PABAK
Hate/Non-hate	0.862	0.868
Targets of Hate		
Race	0.888	0.997
Physical	0.765	0.985
Sex	0.592	0.859
Disability	0.569	0.992
Religion	0.666	0.997
Class	0.431	0.949
Quality	0.549	0.872

While Fleiss' κ penalizes very high or low prevalence of a certain label in a specific target of hate [53], PABAK makes the necessary adjustment to address this [54]. This explains the high PABAK for certain classes despite the low Fleiss' κ. Overall, these values point to an acceptable inter-annotator agreement. With the gold standard as the guide for quality control of judgments, subsequent tweets were tagged, each receiving independent labels from two researchers. Only those with unanimous labeling were included in the final dataset, which is quantitatively described in Table 3.

Table 3. Dataset Composition

Label	Number of Tweets
Hate	8,600 (46.58%)
Race	75
Physical	858
Sex	109
Disability	40
Religion	29
Class	222
Quality	7,641
Non-hate	9,864 (53.42%)
Total	18,464 (100.00%)

3.3 Data Preprocessing

Data preprocessing was performed to prepare the tweets for feature extraction and classification. It consisted of the following steps: data de-identification, uniform resource locator (URL) removal, special character processing, normalization, hashtag processing, and tokenization.

3.3.1 Data De-identification

In order to protect user privacy, the tweets were de-identified through the removal of handles and email addresses using regular expression (regex). The regex for the removal of email addresses was written following the formal definition given in the RFC 5322 standard [55].

3.3.2 Uniform Resource Locator Removal

Although the presence of URLs was investigated by Anzovino, Fersini, and Rosso [53] as a potential cue for detecting misogynistic derailing, they were removed in the present research using regex; since this study focused on the application of language-independent surface features (term frequency-inverse document frequency and term occurrence), URLs were not deemed contributive to building generalizable hate speech classifiers.

3.3.3 Special Character Processing

Special (i.e., non-alphanumeric) characters were removed with the following exceptions, which may represent valuable features in hate speech recognition:

- Those found in expletives (e.g., *t*ngin@)*
- Those constituting emotion-carrying punctuation strings (e.g., *??!!!*): The presence of an exclamation point (used to express intense feelings or sentiments), question mark (used in interrogative statements), and/or swung dash (may be used to signal a playful tone) was taken as an indicator for emotion-indicative punctuation strings.
- Those constituting emoticons (e.g., *:)))))))*): Emoticons were identified with the use of a manually-compiled list.

3.3.4 Normalization

Since posts on social media sites tend to have noisy and malformed texts alongside words and strings with little to no predictive power, an integral part of data cleaning is normalization, which was done in this research through the:

- Removal of numeric characters
- Escaping of hypertext markup language (HTML) characters (using Beautiful Soup [56], an HTML document parser for Python) and removal of resulting punctuation marks that are not listed in the exceptions above
- Removal of diacritics (using Unidecode [57], a Python module that takes Unicode data and converts them into their nearest universally-displayable equivalent)
- Conversion of letters to lowercase
- Removal of stop words, the names of the candidates for the 2016 Philippine elections, and the string *rt* (retweet)

Both English and Filipino stop words were removed. English stop words followed the built-in Natural Language Processing Toolkit [58] list. Filipino stop words included pronouns (along with their shorthand and ligatured forms), determiners, copulatives, and select prepositions and conjunctions. Words that signal disagreement or negation (namely, the Filipino words *di* and *hindi* and the English words *not*, *no*, and *against*) were excluded from the stop word list.

3.3.5 Hashtag Processing

Both the removal and segmentation of hashtags were experimented on with in order to measure their effect on classifier performance. Removal was executed with the use of regex. On the other hand, segmentation was done manually; a total of 1,377 hashtags were segmented. In replacing the original with their segmented forms, the Python library FlashText [59], which clocks in a significantly faster time compared to regex in replacing keywords, was used.

3.3.6 Tokenization

The cleaned tweets underwent tokenization to break them into their constituent words. Since the Natural Language Processing Toolkit Whitespace Tokenizer [58] was used for this process, whitespaces were added before and/or after preserved special character strings (wherever necessary) in order for the tokenizer to recognize them as individual tokens.

3.4 Feature Extraction

Since the collected tweets contained Filipino and English words, mingled with foreign expressions that have entered Filipino lexicon (e.g., the Spanish *que horror*, which translates to *how terrible*) language-independent features were selected for extraction. In particular, surface lexicon-based features were considered: term frequency–inverse document frequency (TF-IDF), term occurrence (TO), and their combination (TF-IDF+TO).

3.4.1 Term Frequency-Inverse Document Frequency

Term frequency-inverse document frequency captures the "weight" or relevance of a term with respect to a document and to the whole document collection by taking the product of term frequency (TF) and inverse document frequency (IDF). While TF is the ratio of the number of times a term appears in a document to the total number of terms in that document, IDF rewards the rarity or uniqueness of a term by computing for the common logarithm of the ratio of the number of documents in which the term appears to the total number of documents in the collection.

Given a term *t*, document *D*, and collection *C*, *N(t, D)* denotes the number of times *t* appears in *D*; *N(D)*, the total number of terms in *D*; *N(t, C)*, the number of documents in *C* containing the term *t*; and *N(C)*, the total number of documents in *C*. The TF-IDF of *t* with respect to *D* and *C*, represented as *TFIDF(t, D, C)*, is formally defined as:

$$TFIDF(t, D, C) = \frac{N(t,D)}{n(D)} \cdot \log\left[\frac{N(t,C)}{N(C)}\right] \qquad \text{(Equation 3)}$$

This research kept the default settings implemented in Scikit-learn [60] as regards TF-IDF smoothing, which prevents zero division through the addition of a document containing all terms, and L2 normalization, which calculates the Euclidean norm $\|x\|_2$ to give the length of the vector $\vec{x} = (x_1, x_2, x_3, \ldots, x_n)$ as mathematically defined by the following equation:

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + x_3^2 + \cdots + x_n^2} \qquad \text{(Equation 4)}$$

For the purpose of keyword analysis, two document-term matrices were constructed. For both matrices, *term* refers to a token unigram extracted from a single tweet and *collection* refers to the entire set of tweets gathered. Two *documents* were considered for the first document-term matrix: the set of non-hate-containing tweets and the set of hate-containing tweets. Meanwhile, for the second matrix, a total of seven documents were taken into account, corresponding to the seven targets of hate.

3.4.2 Term Occurrence

Term occurrence gives a binary representation of the presence of a term in a particular document. Hence, the term occurrence *TO(t, D)* of a term *t* with respect to document *D*, with *N(t, D)* denoting the number of times *t* appears in *D*, is defined as:

$$TO(t, D) = \begin{cases} 0, & N(t, D) = 0 \\ 1, & N(t, D) \geq 1 \end{cases} \quad \text{(Equation 5)}$$

3.5 Classifier Building

After subjecting the dataset to a 70%-30% train-test split, binary and multilabel classifiers were built to classify the tweets into hate- or non-hate containing, and depending on the target(s) of hate, respectively. Three approaches were explored: rule-based, machine learning, and deep learning. In order to reduce imbalance, majority classes were undersampled as shown in Table 4.

Table 4. Dataset Composition with Class Imbalance Reduced

Label	Number of Tweets
Hate	8,600
Race	75
Physical	230
Sex	109
Disability	40
Religion	29
Class	164
Quality	369
Non-hate	8,600

3.5.1 Rule-Based

The rule-based classifier was designed to classify a tweet based on the presence of a keyword. The keyword list was constructed during the annotation process. It included hate-signaling words contained in the tweets and was expanded through the inclusion of related and synonymous words. Sample keywords are presented in Table 5.

Table 5. Sample Keywords Used by the Rule-Based Classifier

Label	Keywords
Hate	*The keyword list was constructed by combining the keyword list for the seven targets of hate.*
Race	amgirl, intsek, kano, makachina, neggy, negneg, negro, nigga, nigger, nigguh
Physical	alien, bansot, boomsunog, dwende, eggnog, kokey, kuba, oily, pandak, pango, panot
Sex	bayot, eutin, iyot, iyutin, kantot, kantutin, manyakis, pokpok, supot, syokla
Disability	abno, abnoy, baliw, flid, lunatic, medicalmysteries, otis, otistik, retard, tarded
Religion	bigot, cult, cultist, devil, devils, kulto, lucifer, osama, satan, satanas
Class	beggar, burgis, elistista, exploiting, greaser, peasant, pleb, pobre, pulubi, timawa
Quality	arogante, bastos, boysawsaw, bullshit, desperate, kingama, korap, tanga, quingina, sinungaling

In particular, a tweet was deemed hate-containing if and only if at least one of the listed keywords was present. Equivalently, a tweet was deemed non-hate-containing if and only if none of the listed keywords were present. This was aligned with the binary nature of the classification task. Meanwhile, a hate-containing tweet was tagged with a target of hate if and only if at least one of the target-specific keywords was present. In light of the multilabel scheme for the second level of classification, a tweet may have multiple targets.

3.5.2 Machine Learning

Using Scikit-learn [60], an open-source machine learning (ML) library for Python, the following ML algorithms were employed to build classification models:

- Linear, nu, and C support vector classifier (SVC)
- Logistic regression
- Multinomial, Bernoulli, and complement naïve Bayes (NB)
- Nearest centroid
- Passive aggressive
- Perceptron
- Random forest
- Ridge regression
- Stochastic gradient descent (SGD)
- XGBoost [61]

The multilabel classification problem was transformed through the method of binary relevance, which decomposes the task by training an independent binary classifier for each label, using the Python library Scikit-multilearn [62], which is built on top of Scikit-learn. While this approach takes less training time, a major limitation is its failure to capture dependencies or relationships between labels; nevertheless, it has been demonstrated to outperform more complex multilabel classification methods in certain experiments [63].

In order to determine the best parameters to optimize the F1 score of the classifiers, hypertuning via stratified ten-fold cross-validated grid search was conducted. Ablation experiments were also done, with the number of TF-IDF features incrementally reduced by 15% (unlike TF-IDF, TO, being a binary representation, is not scalable).

3.5.3 Deep Learning

With the deep learning library Keras [64] running on top of the computational framework TensorFlow [65], a feedforward neural network (FFNN) was designed for binary classification, following the architecture described in Table 6.

Table 6. Feedforward Neural Network Architecture

Layer	Activation Function	Number of Nodes
Input Layer	-	Number of features
Hidden Layer 1	ReLU	1,000
Dropout Layer 1	-	-
Hidden Layer 2	ReLU	500
Dropout Layer 2	-	-
Hidden Layer 3	ReLU	50
Dropout Layer 3	-	-
Output Layer	Softmax	2

The activation function used for the hidden layers was rectified linear unit (ReLU), which works by thresholding values at 0 as defined in Equation 6. The simplicity of the required operations reduces computational complexity and training time (especially in comparison to tanh and sigmoid functions). ReLU also suffers less from vanishing gradients and results to faster convergence.

$$\phi(x) = \max\{0, x\} \quad \text{(Equation 6)}$$

The use of softmax as the activation function for the output layer forces the assignment of discrete probability values with respect to each of the classes instead of simple numeric ones. Formally, given an index $j = 1, 2, 3, \dots, n$ and a vector $\vec{x} = (x_1, x_2, x_3, \dots, x_n)$, the softmax function is defined as:

$$\phi(x)_j = \frac{e^{x_j}}{\sum_{i=1}^{n} e^{x_i}} \quad \text{(Equation 7)}$$

The FFNN was trained over five epochs, with dropout rate set to 50% to prevent overfitting. The loss function was sparse categorical cross entropy, and the optimization algorithm utilized was adaptive moment estimation (Adam), with the default parameters kept at $\alpha = 1 \times 10^{-3}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1 \times 10^{-8}$, and no decay.

3.6 Performance Evaluation

The performance of the built binary classifiers was evaluated using precision, recall, F1, and accuracy scores:

$$\text{precision} = \frac{|\text{true positive}|}{|\text{true positive}|+|\text{false positive}|} \quad \text{(Equation 8)}$$

$$\text{recall} = \frac{|\text{true positive}|}{|\text{true positive}|+|\text{false negative}|} \quad \text{(Equation 9)}$$

$$\text{F1} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad \text{(Equation 10)}$$

$$\text{accuracy} = \frac{|\text{true positive}|+|\text{true negative}|}{\text{total number of labels}} \quad \text{(Equation 11)}$$

Meanwhile, for multilabel classification, the selected metrics were micro-precision, micro-recall, and micro-F1 scores, and Hamming loss. Micro-averaged metrics are calculated globally, placing equal importance on each instance. If the classification task involves *K* classes and $|\text{true positive}|_k$ pertains to the number of true positive labels with respect to class *k* ($|\text{true negative}|_k$, $|\text{false positive}|_k$, and $|\text{false negative}|_k$ are similarly defined), the micro-averaged metrics are then given by the following equations:

$$\text{precision}_{\text{micro}} = \frac{\sum_{k=1}^{K}|\text{true positive}|_k}{\sum_{k=1}^{K}|\text{true positive}|_k+\sum_{k=1}^{K}|\text{true positive}|_k} \quad \text{(Equation 12)}$$

$$\text{recall}_{\text{micro}} = \frac{\sum_{k=1}^{K}|\text{true positive}|_k}{\sum_{k=1}^{K}|\text{true positive}|_k+\sum_{k=1}^{K}|\text{false negative}|_k} \quad \text{(Equation 13)}$$

$$\text{F1}_{\text{micro}} = 2 \cdot \frac{\text{precision}_{\text{micro}} \cdot \text{recall}_{\text{micro}}}{\text{precision}_{\text{micro}}+ \text{recall}_{\text{micro}}} \quad \text{(Equation 14)}$$

Unlike the other metrics, Hamming loss is a loss function that may be related to accuracy in a multilabel setting; hence, a lower value that is closer to zero actually corresponds to a better performance. Compared to 0-1 loss, Hamming loss is a more relaxed metric since it does not require a fully-correct prediction, or a perfect match between the sets of annotated and predicted labels:

$$\text{Hamming loss} = \frac{|\text{false positive}|+|\text{false negative}|}{\text{total number of labels}} \quad \text{(Equation 15)}$$

The macro-F1 scores (which place equal weight on each class) of the multilabel classifiers with the highest micro-F1 scores were computed to provide a quantitative measure of their performance with respect to underrepresented classes. Macro-averaged F1 is the arithmetic mean of the F1 scores for each class, represented as F1_k:

$$\text{F1}_{\text{macro}} = \frac{1}{K}\sum_{k=1}^{K}\text{F1}_k \quad \text{(Equation 16)}$$

4. RESULTS AND DISCUSSION

In this section, the results of the analysis of keywords, evaluation of classifier performance, and application of the top-performing classifier on the entire corpus of 2016 Philippine election-related tweets gathered by Andrade *et al.* [24] are discussed.

4.1 Keyword Analysis

After repeated trials by increments of 5%, it was found that taking the top 40% keywords by TF-IDF yielded the highest percentage of keywords unique to each document as reported in Table 7. Hashtag segmentation did not affect the percentage of unique keywords by more than 5%.

Table 7. Percentage of Unique Keywords

Document	Hashtags Removed	Hashtags Segmented
Hate	52.61%	51.80%
Non-hate	53.00%	52.47%
Targets of Hate		
Race	42.20%	45.66%
Physical	33.68%	34.06%
Sex	52.49%	55.60%
Disability	43.56%	48.00%
Religion	61.63%	56.47%
Class	38.07%	39.94%
Quality	83.78%	84.28%

The top ten keywords by TF-IDF, which are generally the same regardless of hashtag treatment, are enumerated in Table 8. It can be seen that the keywords which are common to both hate- and non-hate-containing tweets include:

- Punctuation strings that are indicative of strong emotions, albeit not necessarily hateful (e.g., *?????*)
- Emoticons that are not necessarily hateful (e.g., *:()*
- Function words that indicate negation (e.g., *no*)
- Offensive words that frequently appear in non-hateful contexts, such as news leads (e.g., *corrupt*)
- Non-offensive words that are associated with political biases (e.g., *dilaw* [Tagalog for *yellow*], an identifying color of one of the major political parties in the Philippines)
- Election- and campaign-related words that carry neither positive nor negative sentiment (e.g., *commercial*)

Table 8. Top Ten Keywords by TF-IDF

Document	Keywords
Hate (Unique)	desperado, hambog, tanga, puta, fuck, punyeta, annoying, gago, nakakainis
Non-hate (Unique)	pht, ppcrv, trndnl, brigada, update, ca, ateneo, visit, lando, cheapen
Hate and Non-hate (Common)	!, ?, ??, lang, nognog, no, wag, commercial, ad, ????
Targets of Hate	
Race	makumpara, hearings, enebenemyern, ita, nigga, amgirl, collaborator, ikinakahiya, iresponsable, kano
Physical	touches, oily, pumuti, nasunog, madilim, dilim, tigas, kakulay, pinagkaiba, costume
Sex	beb, sex, libog, matuwad, bayot, chicks, tangap, womanizer, kantot, woke
Disability	anomaly, virus, pobya, medical, panic, kesyo, differentiate, ingrown, brave, aketch
Religion	commandments, prejudging, atheist, religious, bible, ten, vizconde, violated, talab, mapataob
Class	mayayaman, elitista, milyong, richer, admits, nahihirapan, noontime, hasyenda, entering, gagastusan
Quality	dayaang, corruption, end, till, desperate, bobo, desperado, results, bad, dayaan

Table 9. Manual Analysis of Top 40% Keywords by TF-IDF

Document	Keyword Analysis	Sample Keywords	Frequency (%)	
			Hashtags Removed	Hashtags Segmented
Hate	Hateful	desperado, hambog, tanga, puta, fuck	33.94%	33.87%
	Non-hateful	prowomen, nagsasalita, magkano, mata, pondo	62.63%	62.57%
	Ambiguous	luh, lmao, sus, huy, lolz	2.56%	2.70%
	Punctuation String	-.-, ./., -_____-, >.<, :-P	0.87%	0.86%
Targets of Hate				
Race	Hateful, Related to Target	nigga, amgirl, kano, neg, chinks	13.92%	12.33%
	Hateful, Unrelated to Target	ikinakahiya, iresponsable, naknampoodle	12.66%	15.07%
	Non-hateful, Related to Target	ita, japanese, asian, capiznon, interracial	10.13%	13.70%
	Non-hateful, Unrelated to Target	distinction, asks, mapansin, belong, souls	63.29%	58.90%
	Ambiguous	-	0.00%	0.00%
	Punctuation String	-	0.00%	0.00%
Physical	Hateful, Related to Target	oily, nasunog, ingkanto, butete, nogskie	11.41%	12.20%
	Hateful, Unrelated to Target	dafuq, binatbat, burat, nyare, mofo	6.38%	6.44%
	Non-hateful, Related to Target	touches, kayumanggi, kulot, tyan, apechotties	12.42%	13.90%
	Non-hateful, Unrelated to Target	pinagkaiba, niall, budhi, background, malayong	66.78%	65.42%
	Ambiguous	hahahahahahahahahaha, inpernes, charrr, wahhaha	2.35%	1.69%
	Punctuation String	??????????????????????, :-D, D:<	0.67%	0.34%
Sex	Hateful, Related to Target	matuwad, bayot, chito, puzzy, manyakis	13.89%	13.14%
	Hateful, Unrelated to Target	nilalangaw, beangry, barangayutakan, eassytohl	4.17%	0.73%
	Non-hateful, Related to Target	sex, bra, viagra, threesome, sexual	20.83%	20.44%
	Non-hateful, Unrelated to Target	woke, nkaupo, brgy, serious, principal	59.72%	64.23%
	Ambiguous	dejoke, tahahaha	1.39%	1.46%
	Punctuation String	-	0.00%	0.00%
Disability	Hateful, Related to Target	nababaliw, mongoloid, retarded	8.33%	9.09%
	Hateful, Unrelated to Target	kesyo, fckn, eepal, magisa	10.42%	9.09%
	Non-hateful, Related to Target	virus, pobya, medical, ingrown, dementia	20.83%	20.45%
	Non-hateful, Unrelated to Target	differentiate, brave, computer, lumitaw, diretso	56.25%	56.82%
	Ambiguous	grabeee	2.08%	2.27%
	Punctuation String	p@#$%&*-+	2.08%	2.27%
Religion	Hateful, Related to Target	satanas, sumanib, devils, cult	6.25%	5.67%
	Hateful, Unrelated to Target	mapataob, againsts, saklap, mofos, mangungurakot	12.50%	13.21%
	Non-hateful, Related to Target	commandments, prejudging, religious, bible, mary	29.17%	24.53%
	Non-hateful, Unrelated to Target	ten, talab, nakain, perhaps, clan	50.00%	54.72%
	Ambiguous	olryt	2.08%	1.89%
	Punctuation String	-	0.00%	0.00%
Class	Hateful, Related to Target	elitista, oligarch, elitists, elitism, serfdom	5.76%	5.22%
	Hateful, Unrelated to Target	walanjong, karumaldumal, kulong, langyang, mob	10.07%	9.70%
	Non-hateful, Related to Target	mayayaman, properties, wealth, farm, nike	25.18%	25.37%
	Non-hateful, Unrelated to Target	noontime, entering, hanga, maghanap, iendorso	54.68%	55.22%
	Ambiguous	haahahahaha, hahhaaha, ayyy	2.16%	2.24%
	Punctuation String	$_$_$_$, !??????????????????, xo	2.16%	2.24%
Quality*	Hateful	dayaang, desperate, bobo, desperado, epal	19.80%	19.47%
	Hateful, Related to Other Targets	maniac, pagmumukha, kultong, squatters, budoy	0.13%	0.16%
	Non-hateful	gives, wishing, bunga, betwn, inevitability	77.62%	77.85%
	Ambiguous	pls, seriously, ewan, laughtrip, tbh	1.72%	1.73%
	Punctuation String	-_-, :3, :/, ;(, !!????	0.73%	0.79%

** The scheme for quality is different since it serves as the catchall category for tweets that cannot be classified under any of the preceding targets.*

A manual analysis of the computer-generated keywords for hate showed that they can be categorized into hateful, non-hateful, and ambiguous words, and punctuation strings (including emoticons). *Ambiguous* words, such as *sorry*, pertain to those that may be used both to express non-hateful emotions (as in *We are sorry for the inconvenience this may have caused you)* and to make deriding or sarcastic remarks *(Sorry na lang ang pangit mo kasi talaga)*.

The same coding scheme was extended for the different targets of hate, with the addition of determining whether the keyword is actually pertinent to the target. The quantitative results, along with sample keywords, are reported in Table 9.

It was found that majority of the keywords were non-hateful, with the hateful tokens consistently tallying up to less than a quarter of the entire set of keywords. This is not unexpected since, in real-life conversations and discourses, the frequency of hateful utterances is significantly smaller than that of non-hateful or neutral expressions.

With regard to the targets of hate, the number of related non-hateful keywords consistently outweighed that of related hateful keywords. A possible reason is that only a fraction of tweets actually contained a hate-expressing keyword specific to a target; most constructions involved a neutral word referring to the target, which was then adjoined to a general hateful modifier (an adjective or an adverb).

It is also to be noted that some keywords, especially those that are of a sexual nature, snidely referenced non-election-related events or occurrences (e.g., news about celebrities, television shows, and Internet videos) that trended at the same time as the electoral campaign period. Sardonic word plays on the names of candidates and their political slogans were also prominent; these call to mind a remark made by Speier on the subject of political humor:

> Since a man's name is felt to be a constitutive part of a person, something that is true both in primitive and contemporary cultures, jokes that disfigure or make sport of a name are especially aggressive. They kill in a magical way. [66]

4.2 Performance Evaluation

In this section, the F1 scores (which give the true predictive power by taking the harmonic mean of the precision and recall scores) and the accuracy scores (for binary classification) or Hamming losses (for multilabel classification) of the rule-based, machine learning, and deep learning classifiers are reported and compared.

4.2.1 Rule-Based Classifiers

As reported in Table 10, the rule-based binary classifier performed better than random guessing, with hashtag segmentation increasing the F1 score by 0.94% (from 70.18% to 71.12%) and the accuracy score by 0.51% (from 75.52% to 76.03%).

With respect to the rule-based multilabel classifiers, the increase in micro- and macro-F1 scores were limited to 0.88% (from 59.67% to 60.55%) and 0.31% (from 45.84% to 46.15%). These figures show that, for both classification tasks, hashtag segmentation was unable to improve performance by above 1%.

Table 10. F1 and Accuracy Scores of the Rule-Based Classifier

Label	F1		Accuracy/Hamming Loss*	
	Hashtags Removed	Hashtags Segmented	Hashtags Removed	Hashtags Segmented
Binary	70.18%	71.12%	75.52%	76.03%
Multilabel				
Race	42.67%	42.67%	5.80%	5.80%
Physical	75.37%	75.98%	12.89%	12.63%
Sex	41.73%	41.73%	10.44%	10.44%
Disability	73.02%	73.02%	2.19%	2.24%
Religion	17.65%	17.65%	3.61%	3.61%
Class	1.21%	1.21%	21.01%	21.00%
Quality	69.23%	70.76%	26.80%	25.77%
Micro-F1	59.67%	60.55%		
Macro-F1	45.84%	46.15%		

** Accuracy score is a metric to measure the performance of the binary classifier while Hamming loss is used for the multilabel classifiers. Better performance is signified by an accuracy score closer to 1 and a Hamming loss closer to 0.*

Error Analysis. Analyzing misclassifications shows a prevalence in the number of false negative predictions, as seen in the confusion matrix (Figure 2). Moreover, the number of true predictions can be described as skewed; the classifier is unable to recognize hateful tweets as effectively as it detects non-hateful tweets.

This increase in false negative predictions and the decrease in true positive predictions, as well as the low F1 scores for class and religion, can be attributed to the shortcoming of the keyword-matching operation of the rule-based classifier to account for tweets that express hate but in a manner that does not include an explicit mention of hateful words.

The insufficiency of any keyword list construction to include all hateful utterances, along with their noncanonical spelling variants, also pose a major limitation, considering the complex morphology of Filipino in particular and the dynamicity of language in general.

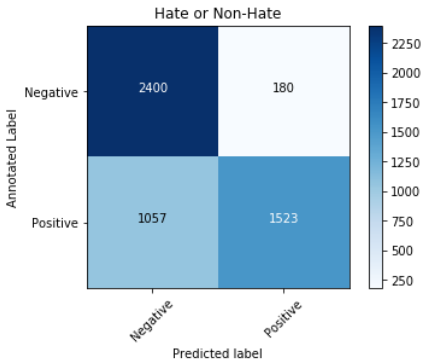

Figure 2. Confusion Matrix for the Rule-Based Binary Classifier with Hashtag Segmentation

4.2.2 Machine and Deep Learning Classifiers

The feedforward neural network model and most machine learning classifiers (with the exception of nearest centroid for the binary classification task and Bernoulli and complement naïve Bayes for the multilabel classification task) fared better than their rule-based counterparts. The F1 scores of these models vis-à-vis the number of features and hashtag treatment are shown in Figures 3 to 7; the graphs on the left side pertain to hashtag removal while those on the right are related to hashtag segmentation.

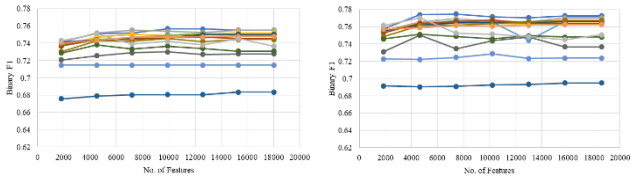

Figure 3. Binary F1 Scores (TF-IDF)

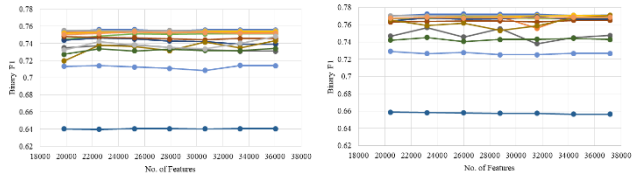

Figure 4. Binary F1 Scores (TF-IDF+TO)

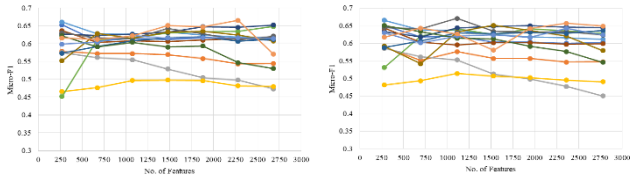

Figure 5. Micro-F1 Scores (TF-IDF)

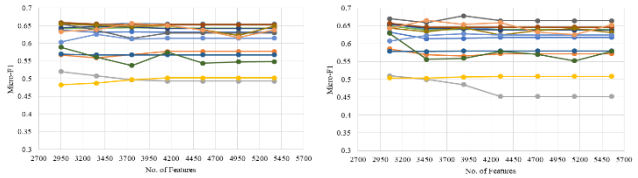

Figure 6. Micro-F1 Scores (TF-IDF+TO)

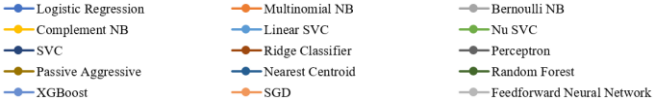

Figure 7. Legend for Figures 3 to 6

Table 11. F1 and Accuracy Scores of Top-Performing Machine and Deep Learning Binary Classifiers*

Classifier	TF-IDF				TO				TF-IDF+TO			
	Hashtags Removed		Hashtags Segmented		Hashtags Removed		Hashtags Segmented		Hashtags Removed		Hashtags Segmented	
	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy
Logistic Regression	**75.68%** **(9907)**	76.10% (9907)	**77.47%** **(7432)**	77.85% (7432)	75.46% (18014)	76.28% (18014)	**77.25%** **(18580)**	78.06% (18580)	**75.62%** **(22517)**	76.39% (22517)	**77.22%** **(26012)**	78.04% (26012)
Multinomial NB	75.18% (15311)	73.90% (15311)	76.55% (15793)	75.39% (15793)	75.29% (18014)	73.59% (18014)	76.94% (18580)	75.47% (18580)	75.38% (30623)	73.74% (30623)	77.07% (34373)	75.64% (34373)
Bernoulli NB	75.50% (15311)	74.30% (15311)	77.03% (15793)	75.95% (15793)	**75.50%** **(18014)**	74.30% (18014)	77.03% (18580)	75.95% (18580)	75.50% (19815)	74.30% (19815)	77.03% (20438)	75.95% (20438)
Complement NB	75.18% (15311)	75.62% (15311)	76.55% (15793)	75.39% (15793)	75.29% (18014)	73.59% (18014)	76.94% (18580)	75.47% (18580)	75.38% (33325)	73.74% (33325)	77.07% (34373)	75.64% (34373)
Nu SVC	75.14% (12609)	75.47% (12609)	76.72% (15793)	77.13% (15793)	74.51% (18014)	75.81% (18014)	76.42% (18580)	77.48% (18580)	75.14% (33325)	76.43% (33325)	76.85% (26012)	75.48% (26012)
Ridge Regression	74.69% (12609)	74.98% (12609)	76.77% (10219)	77.19% (10219)	74.21% (18014)	75.08% (18014)	76.24% (18580)	77.15% (18580)	74.82% (19815)	75.52% (19815)	76.49% (34373)	77.44% (34373)
Perceptron	72.99% (9907)	72.67% (9907)	75.01% (4645)	75.54% (4645)	73.56% (18014)	73.97% (18014)	74.71% (18580)	74.81% (18580)	73.85% (25219)	73.85% (25219)	75.66% (23225)	75.31% (23225)
Passive Aggressive	74.54% (9907)	75.29% (9907)	76.73% (15793)	77.44% (15793)	73.97% (18014)	74.55% (18014)	76.01% (18580)	76.51% (18580)	74.16% (30623)	74.88% (30623)	77.05% (37160)	76.45% (37160)
SGD	74.75% (12609)	74.92% (12609)	76.24% (15793)	76.55% (15793)	74.92% (18014)	75.85% (18014)	76.61% (18580)	77.60% (18580)	75.36% (25219)	76.26% (25219)	76.85% (23225)	77.77% (23225)
Feedforward Neural Network	75.08% (4503)	75.41% (4503)	77.03% (4645)	77.36% (4645)	74.43% (18014)	75.10% (18014)	75.44% (18580)	76.76% (18580)	74.83% (36028)	75.25% (36028)	†	†

** In parentheses are the required number of features to train the classifier; in bold is the top-performing model for each feature.*

† *A feedforward neural network was not implemented for binary classification using TF-IDF+TO with hashtag segmentation due to memory limitation.*

Table 12. Micro-F1 and Accuracy Scores of Top-Performing Machine Learning Multilabel Classifiers*

Classifier	TF-IDF				TO				TF-IDF+TO			
	Hashtags Removed		Hashtags Segmented		Hashtags Removed		Hashtags Segmented		Hashtags Removed		Hashtags Segmented	
	Micro-F1	Hamming Loss	Micro-F1	Hamming Loss	Micro-F1	Hamming Loss	Micro-F1	Hamming Loss	Micro-F1	Hamming Loss	Micro-F1	Hamming Loss
Logistic Regression	65.27% (268)	11.16% (268)	65.00% (277)	11.16% (277)	62.47% (2682)	11.34% (2682)	61.22% (2778)	11.65% (2778)	63.75% (2950)	11.16% (2950)	63.20% (3055)	11.28% (3055)
Linear SVC	66.04% (268)	11.16% (268)	66.54% (277)	10.85% (277)	**65.10%** **(2682)**	10.91% (2682)	64.30% (2778)	11.10% (2778)	65.77% (2950)	10.97% (2950)	65.50% (3055)	10.91% (3055)
Nu SVC	64.82% (2682)	10.91% (2682)	64.14% (1944)	12.20% (1944)	63.69% (2682)	11.47% (2682)	62.91% (2778)	11.17% (2778)	64.45% (3754)	11.16% (2950)	65.00% (3055)	11.16% (3055)
C SVC	65.21% (2682)	10.73% (2682)	64.99% (1944)	11.10% (1944)	64.49% (2682)	11.34% (2682)	64.48% (2778)	11.28% (2778)	64.74% (3352)	11.22% (3352)	65.39% (3055)	11.10% (3055)
Ridge Regression	63.67% (2682)	11.40% (2682)	64.29% (277)	11.04% (277)	64.02% (2682)	11.10% (2682)	63.73% (2778)	11.10% (2778)	**65.89%** **(2950)**	10.79% (2950)	65.74% (3055)	10.61% (3055)
Perceptron	64.22% (1475)	11.34% (1475)	**67.05%** **(1111)**	10.67% (1111)	59.00% (2682)	11.58% (2682)	**64.53%** **(2778)**	11.53% (2778)	65.52% (2950)	11.04% (2950)	**67.80%** **(3889)**	10.42% (3889)
Passive Aggressive	63.47% (1877)	11.22% (1877)	65.07% (1527)	11.65% (1527)	64.66% (2682)	11.53% (2682)	63.17% (2778)	11.22% (2778)	65.54% (2950)	11.22% (2950)	64.55% (3889)	10.97% (3889)
Random Forest	62.19% (268)	12.08% (268)	65.04% (277)	11.40% (277)	56.07% (2682)	12.20% (2682)	57.02% (2778)	11.83% (2778)	58.90% (2950)	12.32% (2950)	62.90% (3055)	11.28% (3055)
XGBoost	62.28% (1072)	11.96% (1072)	64.22% (2361)	11.34% (2361)	60.58% (2682)	11.65% (2682)	63.45% (2778)	11.16% (2778)	62.53% (3352)	11.47% (3352)	62.81% (3889)	11.83% (3889)
SGD	**66.55%** **(2279)**	11.59% (2279)	65.67% (2361)	12.63% (2361)	61.41% (2682)	11.40% (2682)	62.53% (2778)	11.47% (2778)	65.63% (3754)	10.85% (3754)	66.55% (3472)	11.28% (3472)

** In parentheses are the required number of features to train the classifier; in bold is the top-performing model for each feature.*

Binary Classification. As seen in Table 11 and in Figures 3 and 4, the performances of the classifiers were close to each other. The use of TF-IDF, TO, or TF-IDF+TO generally did not vary F1 scores by above 1% and accuracy scores by above 3%. Meanwhile, segmenting hashtags increased F1 by around 1% to 3%.

Logistic regression (with C = 5 and SAGA as solver) performed best, yielding F1 scores of 77.47% (TF-IDF, hashtags segmented), 77.25% (TO, hashtags segmented), and 77.22% (TF-IDF+TO, hashtags segmented, outperforming the rule-based classifier by around 6%. They also registered accuracy scores of around 78%, signifying approximately 2% improvement over the peak accuracy of the rule-based classifier.

Among these three logistic regression classifiers, the best model needed the least training time and the least number of features, requiring only the top 40% keywords by TF-IDF. The effectiveness of logistic regression in this binary classification task is also supported by the fact that all the top-performing models for each feature employed the said machine learning technique, with the sole exception of term occurrence with hashtag removal (the best model for which utilized Bernoulli naïve Bayes). On the flip side, nearest centroid consistently performed poorly. It was the only machine learning model that failed to reach an F1 score of 70%, even with hypertuning of parameters.

Albeit unable to outperform the best logistic regression classifier entirely, the deep learning classifier, which followed a feedforward neural network performed competitively, registering peak F1 and accuracy scores of 77.03% and 77.36%, respectively (TF-IDF, hashtags segmented). Interestingly, although neural networks tend to be computationally expensive, the FFNN required the least number of features among the top-performing models, using only the top 25% keywords by TF-IDF.

Multilabel Classification. As in the case of binary classification, the different machine learning models yielded close performances. For most classifiers, the use of TF-IDF outperformed the use of TO

by around 1% to 6% in terms of F1 score, as seen in Table 12 and Figures 5 and 6. A notable outlier is Bernoulli naïve Bayes, which showed above 10% increase. However, comparing TF-IDF against TF-IDF+TO and hashtag segmentation against hashtag removal does not seem to be straightforward, with performance dependent on the classifier. The choice of features and treatment of hashtags also had minimal effect on Hamming loss, varying the said metric by around 1% only.

Perceptron (with $\alpha = 1 \times 10^{-3}$) performed best, registering the following F1 scores: 67.80% (TF-IDF+TO, hashtags segmented) and 67.05% (TF-IDF, hashtags segmented). Stochastic gradient descent (with $\alpha = 1 \times 10^{-3}$) ranked third, giving an F1 score of 66.55% (TF-IDF, hashtags removed). Their Hamming losses were limited to less than 12%. Among these three models, the second top-performing classifier clocked in the least training time, using only the top 40% keywords by TF-IDF.

Table 13. Macro-F1 Scores of the Top Three Machine Learning Multilabel Classifiers

Classifier	Macro-F1
Perceptron (TF-IDF+TO, hashtags segmented)	61.86%
Perceptron (TF-IDF, hashtags segmented)	59.14%
SGD (TF-IDF, hashtags removed)	51.29%

In order to measure their performance with respect to the minority classes, their macro-F1 scores were computed and are reported in Table 13. As seen in this table, the best perceptron model recorded a 15.71% increase in macro-F1 score in comparison to the rule-based classifier, implying an improved performance in predicting minority classes.

While the micro-F1 scores of the best machine learning classifiers were around 7% higher than that of the rule-based classifier, naïve Bayes performed poorly, with the Bernoulli and complement naïve Bayes models failing to outperform their rule-based counterpart.

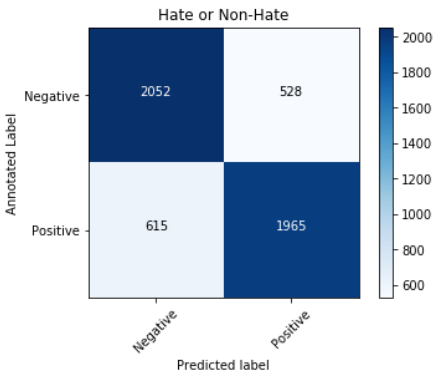

Figure 8. Confusion Matrix for the Top-Performing Logistic Regression Binary Classifier

Error Analysis. As seen in Figure 8, the top-performing machine learning binary classifier, unlike its rule-based counterpart, showed a roughly equal ability in recognizing both hateful and non-hateful tweets. A skewness in the number of true positive predictions was not observed, evincing the effectiveness of using simple lexicon-based, language-independent features.

The false negative and false positive predictions can be traced to the presence of keywords that are common to both hate-containing and non-hate-containing tweets (enumerated in Section 4.1), as well as ambiguous words (examples of which are given in Table 9). Furthermore, since the models only considered token unigrams, spelling variants that need character-level treatment and important phrasal constructions that require longer *n*-grams (for instance, unigrams are unable to capture the essence of the word *not* in the phrase *not corrupt*) contributed to misclassification.

The low F1 score for religion (around 36%) is not unexpected since minority classes, especially those in a heavily-imbalanced dataset as in the case of tagging the targets of hate, are known to pose a problem to machine learning classifiers [67]. This can be ascribed to the difficulty of discerning patterns given the sparsity of the number of samples; in the case of religion, the training and test sets consisted of only twenty and nine tweets, respectively.

4.3 Application on the Entire Corpus

The entire corpus of tweets crawled by Andrade *et al.* [24] was fed to the top-performing classifiers that employed hashtag removal, i.e., logistic regression using TF-IDF (binary F1 = 75.68%) and stochastic gradient descent using TF-IDF (micro-F1 = 66.55% and macro-F1 = 51.29%).

Each presidential candidate was mentioned in a roughly equal proportion of hate-containing tweets, as reported in Table 14. A candidate was tagged based solely on the mention of his/her name or a related hashtag; it is not necessary for the hate to be directed towards the candidate. In addition, the detection and discarding of tweets posted by bots were outside the scope of this study.

Table 14. Proportion of Hate- and Non-Hate-Containing Tweets per Presidential Candidate

Label	1	2	3	4	5
Hate	63166	19508	27657	37165	236893
	(22.39%)	(20.02%)	(21.94%)	(22.93%)	(25.05%)
Race	188	764	185	238	625
Physical	8093	834	1431	565	4317
Sex	819	83	390	734	3435
Disability	39	7	77	14	143
Religion	87	10	154	33	1594
Class	1758	178	422	322	1762
Quality	41484	6142	13987	9402	37356
Non-Hate	218977	77933	98418	124895	708768
	(77.61%)	(79.98%)	(78.06%)	(77.07%)	(74.95%)

The top ten keywords by TF-IDF that are unique in tweets classified as hate-containing are enumerated in Table 15. As seen in this table, there are noticeable intersections in the keywords generated per candidate. In general, the top keywords by TF-IDF include:

- Punctuation strings that are indicative of strong emotions,
- Themes that are associated with a candidate's platform, campaign, background, or characteristics
- Imputations that are specific to a candidate's platform, campaign, background, or characteristics
- Words that are used to express general hatred or contempt, such as expletives
- Function words that serve to intensify, restrict, or compare
- Election- and campaign-related words that carry neither positive nor negative sentiment

Table 15. Top Ten Keywords by TF-IDF Unique in Tweets Classified as Hate-Containing

Candidate	Keywords
A	!, ?, ??, eh, bakit, lang, tatay, panday, sabi, magiging
B	!, ?, ??, lang, rally, ????, grand, president, mayor, talaga
C	!, ?, ??, lang, president, ????, ??????, talaga, no, not
D	!, ?, ??, lang, mas, ????, president, never, talaga, kesa
E	!, ?, ??, vp, lang, laban, tuloy, nognog, talaga, ????

5. CONCLUSION AND FUTURE WORKS

This research was able to show the effectiveness of using simple lexicon-based, language independent-features, specifically term frequency–inverse document frequency, term occurrence, and their combination, in detecting and classifying hate speech in Philippine election-related tweets. It is emphasized though, that the models built were trained and tested on a specific domain of hate speech, that is, hate speech during elections. While the type of features used in our models may be applied to other domains of hate speech, the actual feature values will be different; thus, retraining of models is needed.

As regards the binary classification task, logistic regression using TF-IDF and with hashtag segmentation gave the best performance, outperforming the rule-based classifier by around 6% in terms of F1 score. Although it was unable to entirely outperform the top logistic regression model, the feedforward neural network scored competitively and used fewer features compared to the machine learning models.

Meanwhile, for the multilabel classification task, the top classifier employed perceptron using TF-IDF+TO and with hashtag segmentation, outperforming its rule-based counterpart by 15.71% and 7.25% in terms of macro- and micro-F1 scores, respectively. While predicting minority classes posed a difficulty to the machine learning multilabel classifiers, the binary classifiers yielded acceptable results even with a small dataset.

Future works may focus on extending the methods of this research to hate speech detection in general. The addition of rules aside from lexicon-based keyword matching may improve the performance of the baseline rule-based classifier. In-depth features, such as those that are language-specific or context-aware, may be explored; these include cluster analysis, topic modeling, embeddings, skip-grams, part-of-speech tagging, and metadata extraction. Investigations may also be done to determine and to compare the effectiveness of other machine learning models and deep learning architectures.

6. ACKNOWELDGMENTS

The researchers extend their gratitude to Ms. Charibeth K. Cheng, assistant professor in the Software Technology Department of the College of Computer Studies at De La Salle University, for sharing her time and her expertise as the research adviser. They also thank Mr. Edward P. Tighe, assistant professor in the same department, for his comments and suggestions as their co-panelist, together with Ms. Cheng, during their proposal defense.

7. DISCLOSURE STATEMENT

The researchers declare no potential conflict of interest. In addition, the keywords and examples found in this study are the results of processing actual crawled tweets and in no way reflect the opinion of the authors.

8. FUNDING

No external funding was received for this study.

9. REFERENCES

[1] Cohen-Almagor, R. 2014. Countering hate on the internet. *Annual Review of Law and Ethics* 22 (Dec. 2014), 431–443. https://ssrn.com/abstract=2543511.

[2] Amanullah, M. G. and Dwisusilo, S. M. 2018. Post-truth and echo chamber phenomena of Indonesian social media: analysis of political contestation of Jokowi and Prabowo's supporters in Facebook. In *Proceedings of the International Conference on Language Phenomena in Multimodal Communication* (Surabaya, Indonesia, Jul. 17–19, 2018). DOI= https://doi.org/10.2991/klua-18.2018.14

[3] Bruns, A. 2017. Echo chamber? What echo chamber? 2017. Reviewing the evidence. In *6th Biennial Future of Journalism Conference* (Cardiff, United Kingdom, Sept. 14–15, 2017).

[4] Flaxman, S., Goel, S., and Rao, J. M. 2016. Filter bubbles, echo chambers, and online news consumption. *Public Opinion Quarterly* 80, Special Issue (Jan. 01, 2016), 298–320. DOI= https://doi.org/10.1093/poq/nfw006.

[5] Hosanagar, K., Fleder, D., Lee, D., and Buja, A. 2014. Will the global village fracture into tribes? Recommender systems and their effects on consumer fragmentation. *Management Science* 60, 4 (Apr. 2014), 805-823. DOI= https://dx.doi.org/10.1287/mnsc.2013.1808.

[6] De Smedt, T., Jaki, S., Kotzé, E., Saoud, L., Gwóźdź, M., De Pauw, G., and Daelemans, W. 2018. *Multilingual Cross-Domain Perspectives on Online Hate Speech.* Computational Linguistics & Psycholinguistics Technical Report Series, CTRS-008 (Sept. 2018). University of Antwerp.

[7] Walters, M. A., Brown, R., and Wiedlitzka, S. 2016. *Causes and Motivations of Hate Crime.* Equality and Human Rights Commission Research Report 102 (Jul. 2016). University of Sussex.

[8] Roberts, C., Innes, M., Williams, M., Tregigda, J., and Gadd, D. 2013. *Understanding Who Commits Hate Crime and Why They Do It.* Welsh Government Social Research.

[9] Müller, K. and Schwarz, C. 2018. Fanning the flames of hate: social media and hate crime. (Nov. 30, 2018). DOI= http://dx.doi.org/10.2139/ssrn.3082972.

[10] International Foundation for Electoral Systems. 2016. *The Influence of Hate Speech as a Political Tool on the Youth of Kosovo.* (Jul. 2016).

[11] Isola, O. 2018. *Tackling the Problem of Hate Speech During Election in Nigeria.* Policy Brief No. 17. Wilson Center.

[12] Fasakin, A., Oyero, O., Oyesomi, K., and Okorie, N. 2017. Use of hate speeches in television political campaign. In *Proceedings of the 4th International Conference on Education, Social Sciences and Humanities* (Dubai, United Arab Emirates, Jul. 10–12, 2017), 1382–1388.

[13] Bock, J., Byrd-Craven, J., and Burkley, M. 2017. The role of sexism in voting in the 2016 presidential election. *Personality and Individual Differences* 119 (Dec. 2017), 189–193. DOI= https://dx.doi.org/10.1016/j.paid.2017.07.026.

[14] Ikeanyibe, O., Ezeibe, C., Mbah, P., and Nwangwu, C. 2017. Political campaign ang democratisation: interrogating the use of hate speech in the 2011 and 2015 general elections in Nigeria. *Journal of Language and Politics* 17, 2 (Oct. 2017). DOI= https://dx.doi.org/10.1075/jlp.16010.ike.

[15] Mending, M. D. J. and Sangcopan, A. J. 2018. *An Act Defining Hate Speech and Providing Penalties Therefor.* (Jan. 16, 2018).

[16] Republic of the Philippines. 1930. *An Act Revising the Penal Code and Other Penal Laws.* (Dec. 08, 1930).

[17] Santiago, M. D. 2012. *An Act Establishing a Magna Carta for Philippine Internet Freedom, Cybercrime Prevention and Law Enforcement, Cyberdefense, and National Cybersecurity.* (Nov. 12, 2012).

[18] Van Hee, C., Lefever, E., Verhoeven, B., Mennes, J., Desmet, B., and De Pauw, G. 2015. Detection and fine-grained classification of cyberbullying events. In *Proceedings of Recent Advances in Natural Language Processing* (Hissar, Bulgaria, Sept. 7–9, 2015), 672–680. Associated for Computational Linguistics.

[19] Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., and Bhamidipati, N. 2015. Hate speech detection with comment embeddings. In *Proceedings of the 24th International Conference on World Wide Web Companion* (Florence, Italy, May 18–22, 2015), 29–30. Association for Computing Machinery. DOI= http://dx.doi.org/10.1145/2740908.2742760.

[20] Burnap, P. and Williams, M. L. 2015. Cyber hate speech on Twitter: an application of machine classification and statistical modeling for policy and decision making. *Policy & Internet* 7, 2 (Apr. 2015). DOI= https://doi.org/10.1002/poi3.85.

[21] Gambäck, B. and Sikdar, U. K. 2017. Using convolutional neural networks to classify hate speech. In *Proceedings of the First Workshop on Abusive Language Online* (Vancouver, Canada, Jul. 30–Aug. 04, 2017), 85–90. Association for Computational Linguistics.

[22] Badjatiya, P., Gupta, S., Gupta, S., and Varma, V. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion* (Perth, Australia, Apr. 03–07, 2017), 759–760. International World Wide Web Conferences Steering Committee. DOI: https://dx.doi.org/10.1145/3041021.3054223.

[23] Esteves, P. 2016. Social media changers landscape of Phl elections. *The Philippine Star* (May 13, 2016). https://www.philstar.com/headlines/2016/05/13/1583095/social-media-changes-landscape-phl-elections

[24] Andrade, R. J. C., Balajadia, R. C. M., Han, K. J., and Cheng, C. K. 2017. *Analyzing Twitter Data from the 2016 Philippine Presidential Elections*. Undergraduate Thesis. De La Salle University.

[25] Spertus, E. 1997. Smokey: automatic recognition of hostile messages. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence* (Providence, Rhode Island, Jul. 27–31, 1997), 1058–1065. American Association for Artificial Intelligence.

[26] Zhang, Z. and Luo, L. 2018. Hate speech detection: a solved problem? The challenging case of long tail on Twitter. *Semantic Web* 1 (Sept. 2018), 1–21.

[27] Schmidt, A. and Wiegand, M. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media* (Valencia, Spain, Apr. 03–07, 2017), 1–10. Association for Computational Linguistics.

[28] Mehdad, Y. and Tetreault, J. 2016. Do characters abuse more than words? In *Proceedings of the Special Interest Group on Discourse and Dialogue 2016 Conference* (Los Angeles, United States of America, Sep, 13–15, 2016). Association for Computational Linguistics.

[29] Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., and Chang, Y. 2016. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web* (Montréal, Canada, Apr. 11–15, 2016), 145–153. International World Wide Web Conferences Steering Committee. DOI= https://dx.doi.org/10.1145/2872427.2883062.

[30] Xiang, G., Fan, B., Wang, L., Hong, J., and Rose, C. 2012. Detecting offensive tweets via topical feature discovery over a large scale Twitter corpus. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (Maui, United States of America, Oct. 29–Nov. 02, 2012), 1980–1984. Association for Computing Machinery. DOI= https://dx.doi.org/10.1145/2396761.2398556.

[31] Kshirsagar, R., Cukuvac, T., McKeown, K., and McGregor, S. 2018. Predictive embeddings for hate speech detection in Twitter. In *Proceedings of the Second Workshop on Abusive Language Online* (Brussels, Belgium, Oct. 31, 2018). Association for Computational Linguistics.

[32] Le, Q. and Mikolov, T. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning* (Beijing, China, Jun. 21–26, 2014), 1188–1196.

[33] Sood, S. O., Churchill, E. F., and Antin, J. 2012. Automatic identification of personal insults on social news sites. *Journal of the Association for Information Science and Technology* 63, 2 (Feb. 2012), 270–285.

[34] Razavi, A. H., Inkpen, D., Uritsky, S., and Matwin, S. 2010. Offensive language detection using multi-level classification. In *Proceedings of the 23rd Canadian Conference on Advances in Artificial Intelligence* (Ottawa, Canada, May 31–Jun. 02, 2010), 16–27. Springer-Verlag Berlin. DOI= https://dx.doi.org/10.1007/978-3-642-13059-5.

[35] Gitari, N. D., Zuping, Z., Damien, H., and Long, J. 2015. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10, 4 (Apr. 2015), 215–230. DOI= https://dx.doi.org/10.14257/ijmue.2015.10.4.21

[36] Xu, J. M., Jun, K. S., Zhu, X., and Bellmore, A. 2012. Learning from bullying traces in social media. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Montréal, Canada, Jun. 03–08, 2012), 656–666. Association for Computational Linguistics.

[37] Chen, Y., Zhou, Y., Zhu, S., and Xu, H. 2012. Detecting offensive language in social media to protect adolescent online safety. In *Privacy, Security, Risk and Trust, 2012 International Conference on and 2012 International Conference on Social Computing* (Amsterdam, Netherlands, Sept. 2012), 71–80. Institute of Electrical and Electronics Engineers.

[38] Burnap, P. and Williams, M. L. 2016. Us and them: identifying cyber hare on Twitter across multiple protected characteristics. *EPJ Data Science* 5, 1 (Dec. 2016), 1–15. DOI= https://doi.org/10.1140/epjds/s13688-016-0072-6.

[39] Dinakar, K., Jones, B., Havasi, C., Lieberman, H., and Picard, R. 2012. Common sense reasoning for detection, prevention, and mitigation of cyberbullying. *ACM Transactions on Interactive Intelligent Systems* 2, 3 (Sept. 2012), 1–18

[40] Dadvar, M., Trieschnigg, D., Ordelman, R., and De Jong, F. 2013. Improving cyberbullying detection with user context. In *Proceedings of the 35th European Conference on Advances in Information Retrieval* (Moscow, Russia, Mar. 24–27, 2013), 693–696. Springer-Verlag Berlin.

[41] Waseem, Z. and Hovy, D. 2016. Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (California, United States of America, Jun. 12–17, 2016), 88–93. Association for Computational Linguistics.

[42] Hosseinmardi, H., Mattson, S. A., Rafiq, R. I., Han, R., Lv, Q., and Mishra, S. 2015. Detection of cyberbullying incidents on the Instagram social network. Association for the Advancement of Artificial Intelligence. *CoRR*, abs/1503.03909.

[43] Zhong, H., Li, H., Squicciarini, C., Rajtmajer, S. M., Griffin, C., Miller, D. J., and Caragea, C. 2016. Content-driven detection of cyberbullying on the Instagram social network. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence* (New York, United States of America, Jul. 09–15, 2016), 3952–3958. Association for the Advancement of Artificial Intelligence.

[44] Zimmerman, S., Fox, C., and Kruschwitz, U. 2018. Improving hate speech detection with deep learning ensembles. In *Proceedings of the Eleventh International Conference on Language Resources* (Miyazaki, Japan, May 7–12, 2018), 2546–2553. Association for Computational Linguistics.

[45] Zhang, Z., Robinson, D., and Tepper, J. Detecting hate speech on Twitter using a convolution-GRU based deep neural network. In *Proceedings of the 15th European Semantic Web Conference* (Heraklion, Greece, Jun. 3–7, 2018), 745–760. Springer.

[46] Polignano, M. and Basile, P. 2018. HanSEL: Italian hate speech detection through ensemble learning and deep neural networks. In *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian co-located with the Fifth Italian Conference on Computational Linguistics* (Turin, Italy, Dec. 12–13, 2018). CEUR-WS.org.

[47] Gelashvili, T. 2018. *Hate Speech on Social Media: Implications of Private Regulation and Governance Gaps.* Master's Thesis. Lund University.

[48] Ross, B., Rist, M., Carbonell, G., Cabrera, B., Kurowsky, N., and Wojatzki, M. 2016. Measuring the reliability of hate speech annotations: the case of the European refugee crisis. In *Proceedings of the 3rd Workshop on Natural Language Processing for Computer-Mediated Communication* (Bochum, Germany, September 2016), 6–9. DOI= https://dx.doi.org/10.17185/duepublico/42132

[49] Tighe, E. P. and Cheng, C. K. 2018. Modeling personality traits of Filipino Twitter users. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media* (New Orleans, United States of America, Jun. 2018), 112–122. Association for Computational Linguistics. DOI= https://dx.doi.org/10.18653/v1/W18-1115.

[50] Cheng, C. K. and See, S. L. 2006. The revised Wordframe model for the Filipino language. *Journal of Research in Science, Computing and Engineering* 3, 2 (Aug. 2006), 17–23.

[51] Warner, W. and Hirschberg, J. 2012. Detecting hate speech on the World Wide Web. In *Proceedings of the 2012 Workshop on Language in Social Media* (Montréal, Canada, June 7, 2012), 19–26. Association for Computational Linguistics.

[52] Silva, L., Mondal, L., Correa, D., Benevenuto, F., and Weber, I. 2016. Analyzing the targets of hate in online social media. In *Proceedings of the 10th International Conference on Web and Social Media* (Cologne, Germany, May 17–20, 2016), 687–690. Association for the Advancement of Artificial Intelligence.

[53] Anzovino, M., Fersini, E., and Rosso, P. 2018. Automatic identification and classification of misogynistic language on Twitter. *Natural Language Processing and Information Systems*. Springer. DOI= https://doi.org/10.1007/978-3-319-91947-8_6.

[54] Byrt, T., Bishop, J., and Carlin, J.B. 1993. Bias, prevalence, and kappa. *Journal of Clinical Epidemiology* 46, 5 (May 1993), 423–429. DOI= https://doi.org/10.1016/0895-4356(93)90018-V.

[55] Resnick, P. Ed. 2008. *Internet Message Format.* RFC 5322. (October 2008). https://tools.ietf.org/html/rfc5322.

[56] Richardson, L. *Beautiful Soup.* https://www.crummy.com/software/BeautifulSoup/bs4/doc/#

[57] Burke, S. M. & Solc, T. *Unidecode 1.0.23*. https://pypi.org/project/Unidecode/

[58] *Natural Language Toolkit*. https://www.nltk.org/

[59] Singh, V. 2017. Replace or retrieve keywords in documents at scale. *arXiv:*1711.00046v2.

[60] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. 2011. Scikit-learn: machine learning in Python. *Journal of Machine Learning Research* 12 (Oct. 2011), 2825–2830.

[61] *XGBoost.* https://xgboost.readthedocs.io/en/latest/

[62] Syzmański, P. and Kajdanowicz, T. 2018. A scikit-based Python environment for performing multi-label classification. *arXiv:*1702.01460v5.

[63] Luaces, O., Díez, J., Barranquero, J., and Del Coz, J. J. 2012. Binary relevance efficacy for multilabel classification. *Progress in Artificial Intelligence* 1, 4 (Dec. 2012), 303–313. DOI= https://doi.org/10.1007/s13748-012-0030-x

[64] *Keras: the Python deep learning library.* https://keras.io/

[65] *TensorFlow: an end-to-end open source machine learning platform*. https://www.tensorflow.org/

[66] Speier, H. 1998. Wit and politics: an essay on power and laughter. *The American Journal of Sociology* 103, 5 (Mar. 1998), 1352–1401.

[67] Napierala, K. and Stefanowski, J. 2016. Types of minority class examples and their influence on learning classifiers from imbalanced data. *Journal of Intelligent Information Systems* 46, 3 (June. 2016), 563–597. DOI= https://doi.org/10.1007/s10844-015-0368-1