Offensive Language Detection in Nepali Social Media

Nobal B. Niraula*
Nowa Lab
Madison, Alabama, USA
nobal@nowalab.com

Saurab Dulal*
The University of Memphis
Memphis, Tennessee, USA
sdulal@memphis.edu

Diwa Koirala
Nowa Lab
Madison, Alabama, USA
diwa@nowalab.com

Abstract

Social media texts such as blog posts, comments, and tweets often contain offensive languages including racial hate speech comments, personal attacks, and sexual harassments. Detecting inappropriate use of language is, therefore, of utmost importance for the safety of the users as well as for suppressing hateful conduct and aggression. Existing approaches to this problem are mostly available for resource-rich languages such as English and German. In this paper, we characterize the offensive language in Nepali, a low-resource language, highlighting the challenges that need to be addressed for processing Nepali social media text. We also present experiments for detecting offensive language using supervised machine learning. Besides contributing the first baseline approaches of detecting offensive language in Nepali, we also release human annotated data sets to encourage future research on this crucial topic.

1 Introduction

User-generated content on social media and discussion forums has surged with the advent of technology and the availability of affordable mobile devices. Users interact on these platforms with natural language posts and comments on diverse topics. Such interactions may contain toxic comments or posts that are acutely insulting or harmful to other participants. Such content (foul language) typically consists of racial hate speech, personal attacks, and sexual harassment. Detection of inappropriate use of language is, therefore, of utmost importance. It keeps the discussion healthy by eliminating foul language and also enhances the security of the users by suppressing hateful conduct and aggression.

An approach to filter offensive content is to use human experts (e.g. moderators) and manually review the posts or comments as soon as they get posted. However, manual review is almost impractical and cost-prohibitive, especially when the systems having large user bases that generate a stream of content in a short period. In recent years, the computational linguistics and language technology communities are actively working on automating the detection process. Automated effort can prevent foul content from being posted. It can also flag suspicious content so that human experts monitoring the system can initiate corrective actions.

In this paper, we focus on detecting offensive language in Nepali. While numerous studies exist towards automatic detection of offensive content in resource-rich languages such as English (Gitari et al., 2015; Burnap and Williams, 2016; Davidson et al., 2017; Gambäck and Sikdar, 2017; Waseem, 2016) and German (Schneider et al., 2018; Wiedemann et al., 2018; Michele et al., 2018), to our knowledge, there is no prior work available for a resource-poor language Nepali. Some studies have been found for Hindi (Dalal et al., 2014; Bharti et al., 2017) which is written in the same Devanagari script as Nepali. However, due to the differences in vocabulary, grammar, culture, and ethnicity, systems developed for Hindi do not work for Nepali. Therefore, our novel work presented in this paper lays a foundation for detecting offensive content in Nepali.

The key contributions of this paper are listed as follows:

- We characterize the offensive languages commonly found in Nepali social media.
- We release a human labeled data sets for offensive language detection in Nepali social media which is available at https://github.com/nowalab/offensive-nepali.

*These authors contributed equally to this work

Proceedings of the Fifth Workshop on Online Abuse and Harms, pages 67–75

- We prescribe novel preprocessing approaches for Nepali social media text.

- We provide baseline models for coarse-grained and fine-grained classifications of offensive language in Nepali.

2 Related Work

Detection of hate speech and offensive language across multiple languages is ramping up in recent years. This task is typically modeled as a supervised learning problem that requires a set of human-labeled training examples corresponding to different target classes. The target classes are the types of hate speech or offensive language under the study. Schmidt and Wiegand (2017) provides a comprehensive survey of the approaches in several aspects such as the features used, classification algorithms, and data sets and annotations.

As mentioned previously, majority of studies on hate speech and offensive language detection have been conducted in resource-rich languages such as English and German. Such research is further facilitated by recent competitions and shared tasks that make availability of gold training examples. Toxic Comment Classification Challenge by Kaggle[1], for example, provides thousands of human-labeled examples for detecting toxic behaviors in Wikipedia comments. Similarly, First Shared Task on Aggression Identification (Kumar et al., 2018) for Hindi and English, and Germeval (Wiegand et al., 2018) for German provide gold data sets for detecting offensive languages. The former contains 15000 aggression-annotated Facebook posts and comments each in Hindi and English and the latter contains over 8000 human annotated tweets for German.

An example of hate speech detection in English language is by Burnap and Williams (2016) who studied the detection in tweets with different categories: (a) race (ethnicity), (b) disability, (c) religion, and (c) sexual orientation and transgender status. Their data set consisted of 1803 tweets related to sexual orientation with 183 instances of offensive or antagonistic content, 1876 tweets related to race with 70 instances of offensive or antagonistic content, and 1914 tweets related to the disability with 51 instances of offensive or antagonistic content. The authors modeled the hate speech detection as a classification problem, achieving F-measures of 0.77, 0.75, 0.75, and 0.47 for religion, disability, race, and sexual orientation respectively. Davidson et al. (2017) differentiated hate speech from offensive languages. They classified each English tweet into (a) offensive (b) hate speech and (c) None using different classifiers. Thousands of tweets were labeled using CrowdFlower for the training examples. Several classifiers were trained using a one-versus-rest framework in which a separate classifier was trained for each class and the class label with the highest predicted probability across all classifiers was assigned to each tweet. Out of the several classifiers, logistic regression and support vector machine performed the best achieving the overall precision and recall as 0.91 and 0.90 respectively. However, the precision and recall scores for the hate class were low (precision of 0.44 and recall 0.61), suggesting that the classification of hate speech is challenging. Similarly, Gambäck and Sikdar (2017) trained Convolutional Neural Networks using 6655 Twitter hate-speech data-set originally created by Waseem (2016) to classify utterances into (a) Sexism, (b) Racism, (c) Sexism and Racism, and (d) Non-hate speech, achieving an overall precision, recall, and f-measure as 0.7287, 0.7775, and 0.7389, respectively.

Like in English, detecting offensive languages in German language has also been increased recently especially due to the shared tasks at Germeval 2018[2] and Germeval 2019[3]. Germeval 2018 provided 5009 categorized tweets as training data sets and 3532 as test data sets. It offered two tasks : (1) a coarse-grained binary classification with the categories OFFENSIVE and OTHER and (2) a fine-grained classification with the four categories PROFANITY, INSULT, ABUSE, and OTHER. The training data set consists of 66.3% tweets as OTHER, 20.4% as ABUSE, and 11.9% as INSULT, and only 1.4% as PROFANITY. The best performing system in task 1, TUWienKBS (Montani, 2018), received overall precision, recall, and F-measure of 0.71, 0.65, and 0.68 for OFFENSIVE and 0.82, 0.86, and 0.84 for OTHER respectively. The best performing system, uhhLT(Wiedemann et al., 2018), for the fine-grained task (task 2) achieved average precision, recall, and f-measure as 0.56, 0.49, and 0.52, respectively.

[1] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

[2] https://projects.fzai.h-da.de/iggsa/germeval-2018/

[3] https://projects.fzai.h-da.de/iggsa/

The closest work to ours is the study of linguistic taboos and euphemisms in Nepali by Niraula et al. (2020). The authors presented how the offensive contents are formed in Nepali and also created a resource containing a list of common offensive terms in Nepali. However, they have not addressed the detection of offensive content itself.

3 Offensive Language in Nepali Social Media

Hate speech is a communication that disparages a person or a group based on some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic (Schmidt and Wiegand, 2017). Hate speech can have strong cultural implications (Schmidt and Wiegand, 2017) and thus an utterance can be perceived as offensive or not depending on the observer's cultural background. Besides, the distribution of hate speech can be different in different countries. For example, a country with a mix of religions most likely contains more hate speech related to religions than a country having a singly dominant religion. Therefore, in this section, we discuss different kinds of offensive languages that we observed in Nepali social media.

We reviewed several social media posts and comments on Twitter, YouTube, Facebook, Blogs, and News Portals and identified the common hate speech types. We listed the common types in Table 1 with two examples for each. RACIST (OR), SEXIST(OS), and Other Offensive (OO) (e.g. attack to an individual or organization) are the most commonly observed offensive language types in Nepali social media posts. RACIST (OR) and SEXIST (OS) both are specific cases of offensive content. We noticed an enormous amount of offensive content (OOs) that is not SEXIST or RACIST.

We can expect more of RACIST comments because Nepali society is a mix of several ethnic groups, casts and regions (*pahade* - people live in hilly region; *madheshi* - people live in the south; *ethnic groups* - gurung, magar; *casts* - bahun, chhetri, dalit, etc.). The social tensions among these races and ethnic groups are reflected in the posts and comments.

Hate speeches related to gender and religion are also observed. Interestingly, we observed the hate speech towards females the most when compared with males and the third gender. Targets to Hinduism, Islam, Christianity, and Buddhism are the most common hate speech related to religions. Furthermore, several cases of use of swear words, violent rhetoric, and personal attack towards individuals or organizations are also observed. We categorized them as Other Offensive.

3.1 Challenges in Processing Nepali Social Media Text

Social media text in any language is very noisy and contains ad-hoc typos, abbreviations, acronyms, and hashtags that require a significant amount of preprocessing. In addition to these challenges, Nepali natural language processing requires many other issues to be handled. First, the content can be written in four different ways as shown in Table 1: (a) Nepali text in Devanagari script (b) Nepali text in Roman script, pronunciation-based, (c) pure English text, and (d) Mixed script text that contains both Devanagari and Roman scripts. In addition, cases of *Neglish* in which the user switches between Nepali and English languages are also found. Furthermore, some interesting cases of code-switching were also found, mostly among Hindi, Nepali, Maithili, and English: "सही बोला भाई" (Translation: *rightly said brother*), "गुड night" (Translation: *good night*)

Second, even when the script is written in Devanagari (or Roman), there are several orthographic writing issues one has to deal with while processing Nepali natural language text. The same word (such as वोक्सी) can be written in so many different ways in Devanagari (or in Roman) as they are pronounced almost the same (refer to Table 2).

Third, Nepali is morphologically rich and complex. The same base verb, मार्नु (to kill) for instance, have different forms (मार्छु, मार्छस्, मार्छौं, मार्छेस, मारिन्छ, मारिवक्सिन्छ, मारिनेछ, मारिएला, मारेछ, मारिछे, मारेछन्, मारिएको, मरेको, मारिन्छ, नमार, नमार्नु, etc.) depending on gender, number, honor and tense, giving diverse forms for the same base token. Handling this issue is very crucial for processing Nepali text.

Fourth, Nepali is a low-resource language because Nepali natural language processing is in its infancy. There aren't adequate resources available to process the language. For example, there is not even a list of standard vocabulary words available to use. Lemmatization of morphologically rich languages is crucial but currently is not possible for Nepali. There is no reliable public or commercial parts-of-speech tagger available.

S.N	Content	Type
1	Nepali (Devanagari): मासाला पागल भए जस्तो छ! Translation: "massala" it seems he got mad	OO
2	Nepali (Transliterated): sale khate aphu matra educated thhanndo rahexa Translation: "sale" "khate" (pejorative term for people living in urban slum dwellers) thinks he is the only educated	OO
3	Nepali (Devanagari): पागल् बाहुन् Translation: lunatic "bahun" (an upper cast)	OR
4	Nepali (Transliterated): Rajako kaam chhodi kamiko dewali Translation: Going to kami's festival over king's assignment – a traditionally non-tabooed idiom that is considered racist now	OR
5	Nepali (Transliterated): Pothi baseko suhaudaina Translation: It does not suit a woman to raise her voice (sexist idiom)	OS
6	Nepali (Mixed): पैसामा बीक्छन के टी हरू sala Translation: girls get sold with money *sala*	OS
7	Nepali (Transliterated): ma pani bahun hu tara tapaaik ko kuro chhita bujhena Translation: I am also a bramhin, but I am dissatisfied with your words	NO
8	Nepali (Devanagari): यो भालु हो सर Translation: Sir, this is a bear	NO

Table 1: Examples of common offensive languages found in Nepali social media. Note that they could be typed in (a) *Romanized* (2, 4, 5, 7) (b) *Devanagari Script* (1, 3, 8) and (c) *Mixed* i.e. Romanized + Devanagari (6). OO = Other-Offensive, OR = Offensive Racist, OS = Offensive Sexist, NO = Non-offensive

Script	Content
English	mad witch
Romanized - 1	pagal boksi
Romanized - 2	pagal bokshi
Devanagari - 1	पागल वोक्सी
Devanagari - 2	पागल वोक्सि
Devanagari - 3	पागल वोक्सी
Devanagari - 4	पागल बोक्सि
Devanagari - 5	पागल बोक्शी
Devanagari - 6	पागल बोक्शि
Mixed -1	पागल boksi
Mixed -2	पागल bokshi
Mixed -3	pagal वोक्सी

Table 2: Different orthographic forms of writing the text "mad witch"

Fifth, translation of data sets or resources from other languages to Nepali is not straightforward. Commercially available language translation services are poor in translating contents from other languages to Nepali. All of these issues make the processing of Nepali text very challenging.

4 Methodology

In this section, we describe the data collection, data annotation, and our system to detect offensive languages in Nepali text.

4.1 Data Collection

Our goal is to create a labeled data set of hate speech of different types and train machine learning models using it. Since hate speech appears relatively less in social media, annotating a large sample gives just a few offensive contents, making the annotation process very laborious and expensive. To address this problem, researchers apply different strategies to improve the distribution of offensive content Zampieri et al. (2019). Following these strategies, we made a pool of comments and posts from the sources in social media that have higher chances of containing hate speech. Our pool consists of over 15000 comments and posts from diverse social media platforms such as Facebook, Twitter, YouTube, Nepali Blogs, and News Portals.

For Facebook, we first made a list of potentially controversial posts posted to a general audience in open groups and public pages between 2017 and 2019. We then extracted around 7000 comments corresponding to those posts. For Twitter, we followed a bootstrapping approach as done by prior arts (Zampieri et al., 2019). For this, we first created a small list of Nepali words (in both De-

vanagari and Romanized forms) that have higher chances of being used in hate speech. The words themselves are not explicitly offensive but can appear in hate speech depending on the context of their use. For example, the words "बाहुन" (*bahun* - an upper cast in Nepali society) and "भालु" (*bhalu* - bear) are non-offensive by themselves but can appear in offensive contexts. Offensively, *bahun* can be used to insult someone racially based on their cast, and *bhalu* can be used to call someone a prostitute. Using the list of keywords, we performed a targeted search on Twitter and collected about 4000 tweets, approximately 50 tweets per word. These tweets enhanced the pool with diverse and context-sensitive posts. For YouTube, similar to Facebook, we manually created a list of potentially controversial, non-controversial, and neutral videos, and extracted approximately 3500 comments. Video contents are highly engaging. A good length video – especially a controversial one – contained diverse emotions and attributes such as anger, happiness, low and high pitch, etc., and was scrutinized by the viewers. The YouTube video comments also helped to maintain the diversity of data set in the writing form as they were typed in transliterated, mixed, and pure Devanagari font and fulfill our categorical requirements. Besides, they captured the inputs from the diversity of people commenting on the posts. Finally, the rest of the comments, about 500, were gathered from several Nepali blogs and news websites.

Source	**NO**	**OO**	**OR**	**OS**	**Total**
Twitter	1214	802	39	22	2077
Facebook	2313	853	168	27	3361
YouTube	908	846	56	36	1846
Other	117	51	6	3	177
Total	4552	2552	269	88	7462

Table 3: The pool of social medial data set.

4.2 Data Annotation and Data Set

After constructing the pool of comments and posts, we randomized the records for annotation. To ensure the quality, we used two annotators and asked them to annotate each record into four categories: SEXIST, RACIST, OTHER-OFFENSIVE, and NON-OFFENSIVE. We computed the inter-rater reliability (IRR) between each pair of ratings using Cohen's kappa (k) (McHugh, 2012). IRR scores were computed for both fine-grained (considering all four labels) and coarse-grained (offensive or non-offensive) cases. For the coarse-grained, we considered the three offensive categories SEXIST, RACIST, and OTHER-OFFENSIVE as offensive. The Cohen's kappa coefficients obtained for fine-grained and coarse-grained cases were 0.71 and 0.78, respectively, suggesting substantial agreements between the raters. We observed most of the disagreements between human annotators in borderline cases. For example, *Kati milyo Parti bat Dr. Sab lai* (How much/many did you get from the party[4], Dr. Sab?) was marked as offensive by one while non-offensive by the other. This comment could be a personal attack for corruption in certain contexts while non-offensive in some other e.g. receiving compensation or votes. The disagreements were reviewed by the third annotator and resolved on consensus.

	NO	**OO**	**OR**	**OS**	**Total**
Train	3562	1950	218	68	5798
Test	896	486	49	19	1450

Table 4: Training and Testing Data Sets

Additionally, the social media posts and comments often contained personally identifiable information such as person names, organization names, and phone numbers. To anonymize the comments, we replace the person/organization names with unique random yet real person/organization names. Since gender information carries vital linguistic properties in the language, we tried preserving the gender as much as possible during the name replacement process. A name with a known gender (i.e. male or female) is replaced with another random name of the same gender.

The annotators annotated 7462 records altogether. The distribution of the annotation across different categories is presented in Table 3. We removed the duplicated examples from the annotated corpus and performed 80-20 split randomly to create the training and test data sets. The statistics of these data sets are shown in Table 4. To encourage the research community for addressing this important task of offensive language detection in Nepali, we have released these gold data sets at https://github.com/nowalab/offensive-nepali.

[4]Party here specifically refers to political organization

4.3 Preprocessing

As described in Section 3.1, the social media comments and posts came in different forms: comments purely in Devanagari script, transliteration using Roman letters, pure English, or their combinations. In fact, more than 50% of the comments in our pool are written in transliterated or mixed forms. We speculate, due to the ease of writing, this pattern will continue. These observations reiterate the need for text normalization while processing Nepali social media texts. To this end, we consider two different text normalization schemes:
(A) Dirghikaran (Prep_Dir): Because multiple characters have the same sound, inconsistencies appear even for the same word written in Devanagari script. We use the following mappings to normalize the character variants: ि -> ी, ु -> ू, स -> श, ष -> श, व -> ब, उ -> ऊ, ृरी -> ृ, ृरि -> ृरी, इ -> ई, ं -> ँ, न -> ण, ं -> ङ्. This converts the words with different orthographic forms to a normalized form, e.g., किताव, and किताब both map to कीताब. This approach does not affect the tokens that are already transliterated in Romanized form or written in English.
(B) Romanization (Prep_Rom): With this scheme, we convert (transliterate) each Nepali word written in Devanagari script to its Romanized form using a number of rules. This rule-based system takes care of the orthographic variants as well. For instance, it converts all किताव, किताब, कीताब, and कीताव to *kitab*. We could have done the reverse way i.e. converting transliterated text in Romanized form to Devanagari script (e.g. *kitab* -> किताव) but we found that converting Devanagari text to Romanized using the rules is relatively easier. After this preprocessing, all the comments will be in Romanized forms. This powerful preprocessing technique has not been employed in any of the prior arts and is one of our novel contributions in this paper.

4.4 Features

Nepali, as illustrated in Section 3.1, is a morphologically rich language. A verb, for example, can take different forms depending upon gender, number, honor, tense, and their combinations. Therefore, character-based and sub-word features are expected to be useful in classifying offensive languages. For that reason, we considered both word (Unigrams and Bigrams) and character (Character Trigrams) features for our experiments.

4.5 Experiments

We performed experiments to see the effect of preprocessing scheme and classification model, and coarse and fine-grained classification. In all experiments, we reduced the features down to 10000 using KBest algorithm with chi-squared stat.

Prep.	**Non-Offensive**			**Offensive**		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1
A	0.71	0.94	0.81	0.80	0.39	0.52
B	0.71	0.94	0.81	0.81	0.39	0.53
C	0.76	0.94	0.84	0.85	0.51	0.64
D	0.76	0.95	0.84	0.85	0.51	0.64
A	0.78	0.94	0.85	0.84	0.56	0.68
B	0.78	0.92	0.85	0.83	0.58	0.68
C	0.79	0.91	0.84	0.81	0.60	0.69
D	0.79	0.92	0.85	0.83	0.59	0.69
A	0.78	0.93	0.85	0.83	0.57	0.68
B	0.79	0.92	0.85	0.83	0.60	0.69
C	0.79	0.92	0.85	0.81	0.60	0.69
D	0.79	0.93	0.85	0.83	0.61	0.70

Table 5: Effect of prepossessing techniques and features on binary classification. Preprocessing techniques: (A) No Preprocessing (Prep_None) (B) Dirghikaran (Prep_Dir), (C) Romanization (Prep_Rom), and (D) Prep_Dir + Prep_Rom. The **first block** uses word only, the **second block** uses character only and the **last block** uses both word and character features.

Models	**Non-Offensive**			**Offensive**		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1
Baseline	0.74	0.73	0.73	0.57	0.58	0.58
LR	0.80	0.93	0.87	0.86	0.63	0.72
SVM	0.78	0.95	0.85	0.87	0.56	0.68
RF	0.81	0.93	0.86	0.85	0.64	0.73
M-BERT	0.74	0.90	0.81	0.75	0.49	0.59

Table 6: Binary classification using different machine learning models.

4.6 Effect of Preprocessing

We trained a Logistic Regression classifier for binary classification using four different preprocessing schemes: A. No preprocessing (Prep_None), B. Dirghikaran (Prep_Dir), C. Romanization (Prep_Rom), and D. Both Prep_Dir + Prep_Rom, where + means string concatenation. We considered positive examples as the records with OO, OR, and OS from Table 4. This yielded the train data set with 3562 negative and 2236 positive examples and the test data set with 896 positive and 554 negative examples.

We reported the results using the test data in Table 5. The top, middle, and bottom blocks contain the results corresponding to word only, char-

acter only, and both word and character features, respectively. The results in the middle block are significantly better than the results in the top block, demonstrating that character-based features are extremely useful. It is expected because Nepali is morphologically very rich and the social media text is very noisy. Adding both word and character features further slightly improved the results (the bottom block).

Within each block, i.e. given a feature type, the results are better in the order: D > C > B > A, where A is no preprocessing. The preprocessing technique B, "Dirghikaran", improved the performance of the classifier compared to A. But the margin of improvement by C, "Romanization", is typically higher than that by B. It is especially significant when the word only features are used. This is because Dirghikaran only normalizes the terms written in the Devanagari script but it does not transliterate the text. Romanization, however, transliterates the text written in Devanagari script and makes it uniform with other already transliterated user posts. Combining texts using both Romanization and Dirghikaran, marked with D, slightly improved the results over C.

4.7 Coarse-grained Classification

For coarse-grained (i.e. binary) classification, we experimented with four machine learning classifiers that are most often used for offensive language detection. Specifically, we used: (A) **Logistic Regression (LR)**: Linear LR with L2 regularization constant 1 and limited-memory BFGS optimization, (B) **Support Vector Machine (SVM)**: Linear SVM with L2 regularization constant 1 and logistic loss function, (C) **Random Forests (RF)**: Averaging probabilistic predictions of 100 randomized decision trees. (D) **Multilingual BERT (M-BERT)**: Current best performing models for offensive language detection utilize BERT (Devlin et al., 2018) based models (Liu et al., 2019; Mozafari et al., 2019; Baruah et al., 2020). Although there is no BERT model available for Nepali yet, Nepali is included in M-BERT[5] which is trained using the entire Wikipedia dump for each language. We used Hugging Face Transformer library (Wolf et al., 2020) to build the M-BERT classifier.

In addition, we constructed a **baseline** model using the list of Nepali offensive terms collected by Niraula et al. (2020) and is available at GitHub[6]. This data set contains 1078 offensive terms, their transliterated forms, and interestingly their offensiveness scores. The offensiveness score ranges from 1 (slightly offensive) to 5 (absolute offensive e.g. taboo terms). For a given post, our baseline scans for the tokens present in the dictionary and sums the corresponding offensiveness scores. If the sum is 5 or more, it declares the post as offensive.

For baseline and traditional machine learning models (LR, SVM, and RF), as suggested by the experiments in Section 4.6, we chose the Romanization + Dirghikaran preprocessing strategy and both word and character-based features. In addition, we computed and utilized the indicator features, for each post, by scanning the preprocessed tokens and looking them up in the offensive dictionary. As before, we reduced the features using KBest to 10000 for both train and test data sets.

We trained the models and evaluated them using the binary train and test data sets constructed as described in Section 4.6. The evaluation results are presented in Table 6. The baseline model which is based on a dictionary obtained the F_1 scores of 0.58 and 0.73 for offensive and non-offensive categories. All machine learning models performed very well compared to the baseline model. Interestingly, M-BERT model did not perform well compared to the traditional models. This could be because M-BERT model is trained using Wikipedia content which is different from the social media text. Also, the size of Wikipedia for low-resource language Nepali is not huge and thus it is under-represented in the M-BERT model. Logistic Regression and Random Forrest models were the top-performing models, with the latter having a slightly higher F_1 score on the offensive category. For this reason, we chose the Random Forrest classifier for the fine-grained classification which we describe next.

4.8 Fine-grained classification

Fine-grained classification can be done by directly training a multi-class classifier over the labeled training data set. However, we followed the principle proposed by Park and Fung (2017) that performed better for this specific task. Following this, we trained a Random Forrest classifier for coarse-grained classification as in Section 4.7. We trained

[5] https://github.com/google-research/bert/blob/master/multilingual.md

[6] https://github.com/nowalab/offensive-nepali

	None			Other Offensive			Racist			Sexist		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1
RF	0.81	0.93	0.87	0.79	0.64	0.71	0.76	0.32	0.45	0.9	0.05	0.01

Table 7: Results for detecting different offensive categories

another Random Forrest classifier using only the training data set with labels OO (other offensive), OR (offensive racist), and OS (offensive sexist). During testing, we applied the second classifier only to those test records that the first classifier predicted as offensive to get their fine-grained categories. We assigned a non-offensive label (NO) to each test record for which the first classifier predicted as non-offensive.

We reported the experiment results in Table 7. The F_1 scores for Non-Offensive, Other Offensive, Racist, and Sexist were 0.87, 0.71, 0.45, and 0.01 respectively. The lower performance for the sexist category was mainly due to the fewer training examples available for this category compared to the other categories (see Table 4). Gathering these fine-grained labels is a major challenge in the field than obtaining labels with simply offensive and non-offensive (Park and Fung, 2017). This is more evident in the low-resource language like Nepali.

4.9 Error Analysis

Most of the errors were due to the lack of world and contextual knowledge to the classifier and is always a challenge for offensive language detection in any language. For instance, *thamel ma bhalu ko bigbigi* (literal translation: *Abundant bears in Thamel*) is offensive while *jungle ma bhalu ko bigbigi* (literal meaning: *Abundant bears in jungle*) is non-offensive although both of these sentences have the same tokens everywhere except one i.e. *Thamel* vs. *Jungle*. *Thamel* is a famous tourist area in Kathmandu that also has a negative connotation as a brothel and *bhalu* is a contextually offensive term that can mean a *bear* or a *prostitute* depending on the context.

5 Conclusion

In this paper, we presented a systematic study of offensive language detection in Nepali, a topic that has not been explored for this low resource language. We collected diverse social media posts and generated a labeled data set by manually annotating 7248 posts with fine-grained labels. The data set is available at https://github.com/nowalab/offensive-nepali.

We presented different challenges that need to be addressed to process noisy social media posts in Nepali. We proposed three different preprocessing methods and provided detailed evaluations demonstrating their effectiveness on the model performance. We reported detailed experiments for coarse-grained detection of offensive languages using several conventional machine learning and recent deep learning models and features. We also provided a fine-grained classification of offensive comments using a two-step approach for Nepali language.

Our data set and baseline algorithms provide foundation for future research in this area to fight against cyberbullying and hate speech, which has been widespread in recent days. We would like to caution to those who use our work (e.g. data sets and algorithms) to avoid over-reliance on keywords and machine learning models. We remind everyone to keep the context in the forefront, and encourage using human review to the ones flagged by the machine learning systems as offensive, especially in cases of false positives.

Future work includes detecting the targets of the offensive comments, which could be an individual organization/person or a group. Leveraging offensive language data sets from other languages to Nepali, e.g. by translation and transfer learning as done by Sohn and Lee (2019), is another interesting future direction.

Acknowledgments

We would like acknowledge Ms. Monika Shah, professor Dr. Kumar Prasad Koirala, and Mr. Suraj Subedi for their continued support, helpful discussions and encouragements.

References

Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. 2020. Aggression identification in english, hindi and bangla text using bert, roberta and svm. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 76–82.

Santosh Kumar Bharti, Korra Sathya Babu, and Sanjay Kumar Jena. 2017. Harnessing online news for

sarcasm detection in hindi tweets. In *International Conference on Pattern Recognition and Machine Intelligence*, pages 679–686. Springer.

Pete Burnap and Matthew L Williams. 2016. Us and them: identifying cyber hate on twitter across multiple protected characteristics. *EPJ Data Science*, 5(1):11.

Chetan Dalal, Shivyansh Tandon, and Amitabha Mukerjee. 2014. Insult detection in hindi. Technical report, Technical report on Artificial Intelligence, 18.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International Conference on Web and Social Media*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90.

Njagi Dennis Gitari, Zhang Zuping, Hanyurwimfura Damien, and Jun Long. 2015. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10(4):215–230.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 1–11.

Ping Liu, Wen Li, and Liang Zou. 2019. Nuli at semeval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 87–91.

Mary L McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282.

Corazza Michele, Stefano Menini, Arslan Pinar, Rachele Sprugnoli, Cabrio Elena, Sara Tonelli, and Villata Serena. 2018. Inriafbk at germeval 2018: Identifying offensive tweets using recurrent neural networks. In *GermEval 2018*, pages 80–84.

Joaquın Padilla Montani. 2018. Tuwienkbs at germeval 2018: German abusive tweet detection. *Austrian Academy of Sciences, Vienna September 21, 2018*.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2019. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer.

Nobal B Niraula, Saurab Dulal, and Diwa Koirala. 2020. Linguistic taboos and euphemisms in nepali. *arXiv preprint arXiv:2007.13798*.

Ji Ho Park and Pascale Fung. 2017. One-step and two-step classification for abusive language detection on twitter. *arXiv preprint arXiv:1706.01206*.

Anna Schmidt and Michael Wiegand. 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*, pages 1–10.

Julian Moreno Schneider, Roland Roller, Peter Bourgonje, Stefanie Hegele, and Georg Rehm. 2018. Towards the automatic classification of offensive language and related phenomena in german tweets. *Austrian Academy of Sciences, Vienna September 21, 2018*.

Hajung Sohn and Hyunju Lee. 2019. Mc-bert4hate: Hate speech detection using multi-channel bert for different languages and translations. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 551–559. IEEE.

Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*, pages 138–142.

Gregor Wiedemann, Eugen Ruppert, Raghav Jindal, and Chris Biemann. 2018. Transfer learning from lda to bilstm-cnn for offensive language detection in twitter. *Austrian Academy of Sciences, Vienna September 21, 2018*.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language. *Austrian Academy of Sciences, Vienna September 21, 2018*.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. pages 1415–1420.