Multi-channel Convolutional Neural Network for Hate Speech Detection in Social Media

Zeleke Abebaw[1(✉)], Andreas Rauber[2], and Solomon Atnafu[3]

[1] IT Doctoral Program, Addis Ababa University, Addis Ababa, Ethiopia
zeleke.abebaw@aastu.edu.et
[2] Institute of Information Systems Engineering, Technical University of Vienna, Vienna, Austria
rauber@ifs.tuwien.ac.at
[3] Department of Computer Science, Addis Ababa University, Addis Ababa, Ethiopia
solomon.atnafu@aau.edu.et

Abstract. As online social media content continues to grow, so does the spread of hate speech. Hate speech has devastating consequences unless it is detected and monitored early. Recently, deep neural network-based hate speech detection models, particularly conventional single-channel Convolutional Neural Network (CNN), have achieved remarkable performance. However, the effectiveness of the models depends on the type of language they are trained on and the training data size. We argue that the effectiveness of the models could further be enhanced if we use multi-channel CNN models even for under-resourced languages that have limited training data size. This is because the single-channel CNN might fail to consider the potential effect of multiple channels to generate better features, which is not well investigated for hate speech detection. Therefore, in this work, we explore the use of multi-channel CNN to extract better features from different channels in an end-to-end manner on top of a word2vec embedding layer. Tested on a new small-scale Amharic hate speech dataset containing 2000 annotated social media comments, the experimental results show that the proposed multi-channel CNN model outperforms the single-channel CNN models but underperform from the baseline Support Vector Machine (SVM) with an average F-score of 81.3%, 78.2%, and 92.5% respectively. The finding of the study implies that the proposed MC-CNN model can be used as an alternative solution for hate speech detection using a deep learning approach when dataset scarcity is an issue.

Keywords: Social media · Deep learning · Word embedding · Amharic hate speech detection · Single-channel · Multi-channel · Convolutional neural network

1 Introduction

The increasing accessibility of the Internet and social media (e.g., Facebook, Twitter, etc.) has provided people with a plethora of opportunities and advantages including the ability to keep social relationships in the economic, political, and social spheres. As a result, there is an abundance of user-generated online content on social media [1]. The growth of online social media content as a means of free expression has led in changes

M. L. Berihun (Ed.): ICAST 2021, LNICST 411, pp. 603–618, 2022.
https://doi.org/10.1007/978-3-030-93709-6_41

in the economic, political, and social arenas. Despite its democratic nature for free expression, social media has adverse consequences since it is an ideal venue to disseminate hate speech.

Social media users disseminate hate speech against certain social groups, encouraging others to send nasty messages, write harsh critiques, engage in physical assault, and commit hate crimes [2], which has become a global phenomenon. For example, according to the FBI's annual hate crime statistics report[1] for 2019, hate crimes in America increased by 3%, with race-based, religious-based, and gender-based offenses being the most common. Furthermore, during the 2007 Kenyan elections, due to the spread of hate speech, inter-ethnic conflicts and violence 1,300 people were murdered and over 650,000 people were displaced [3].

In recent years, the spread of hate speech and aggressive comments on social media has been observed on Ethiopian related issues, notably during election periods and political unrest [4]. For example, due to the proliferation of hate speech during Ethiopia's 2016 national election, the government was obliged to shut down the Internet and restricted social media sites in order to deescalate tensions among groups [5]. Such acts continue to occur across the country, causing widespread disruption and unrest that impacted the lives of millions, impeded economic activity, closed highways, displaced communities, and even resulted in the deaths of hundreds [6]. Hate speech and hate crimes, in general, poison society by endangering individual rights, human dignity, and equality, increasing social tensions, disrupting public peace and order, and risking peaceful coexistence [7].

Therefore, several actions have been implemented to combat the spread of hate speech on social media. While governments have taken legal action through law enforcement, social media firms have begun to deploy automatic hate speech detection and monitoring systems based on machine learning algorithms. As a result, numerous studies on automatic hate speech detection have been conducted utilizing supervised machine learning techniques with two stages. The first stage is to build features by hand using feature engineering approaches, and the second stage is to choose and apply the best machine learning classifiers available. Handcrafted feature engineering is vital, but it is time consuming and prone to mistakes. Alternatively, methods based on deep learning models have become increasingly popular. This is because, unlike traditional machine learning approaches, deep learning models jointly implement both feature extraction and classification.

Deep neural network models, such as conventional single-channel CNN (SC-CNN), have recently demonstrated good performance in hate speech detection [7–10]. However, a single-channel CNN overlooked the possibility that new features might well be generated through multi-channel techniques. However, this has not been extensively studied for hate speech detection. We observed that utilizing several channels of the CNN model could capture additional features from each channel that would otherwise be overlooked by the max-pooling layer of the single-channel model. Instead, improved features could be generated from the basic CNN model by

[1] https://www.splcenter.org/news/2020/11/16/fbi-reports-increase-hate-crimes-2019-hate-based-murders-more-doubled.

initializing distinct channels from the input embedding layer and taking the max-pooling layer from each channel and concatenating them. Therefore, in this paper, we present a multi-channel Convolutional Neural Network (MC-CNN) model built on top of the word2vec word embedding layer that detects hate speech effectively using just a small-scale Amharic hate speech dataset.

We have selected Amharic as a test case because hate speech detection and monitoring algorithms deployed by social media platforms to guarantee compliance with community standards or posting laws have failed miserably in under-resourced local languages like Amharic. This is because the hate speech detection and monitoring algorithms were trained on posts written in resourceful languages such as English. As a result, social media posts written in under-resourced languages could easily evade the monitoring algorithms. For example, on May 29, 2020, we posted hate speech in English (primarily for experimental purposes) to put the monitoring algorithms to the test. During the process, the algorithm identified the post as hate speech, indicating that we had broken the terms of the user agreement, and issued a warning notice "*You cannot publish or comment for 24 h*"[2]. However, the translated version of the post to Amharic simply bypassed the system with no warning notice. Due to lack of hate speech detection tools and benchmark datasets, such under-resourced local languages that are utilized for day-to-day communication by over 30 million speakers in Ethiopia suffer from the spread of hate speech. Furthermore, the propagation of hate speech might be a contributing factor to the present cyber disputes, which could have an impact on social life at both the individual and national levels [11]. As a result, testing models on such under-resourced local languages can help to build a moderate social media ecosystem. Furthermore, the proposed approach is not language-specific and can be used in other languages as well. The key contributions of the paper thus are:

1. We develop a hate speech detection system that is not reliant on time-consuming manual feature engineering techniques.
2. We compare the performance of a multi-channel CNN model to that of a single-channel CNN model in detecting Amharic hate speech.
3. We develop a baseline hate speech detection dataset for Amharic language.

The remainder of the paper is structured as follows. The next section provides a review of current literatures. Sections 3 and 4 cover dataset construction and classification models. Section 5 describes our model's architecture. The experiments and its discussions are presented in Sect. 6 and Sect. 7. Finally, Sect. 8 summarizes the article and suggests further research.

2 Review of Literatures

2.1 Defining Hate Speech

Although there are no universal agreements on the definition of hate speech, academics and social media firms have developed their own to assist in classifying user comments

[2] https://www.facebook.com/zeleke.ab.

as hate or non-hate. Hate speech, for example, is defined by [12] as "a statement or expression that denigrates a person or individuals on the basis of (alleged) membership in a social group identifiable by qualities such as color, ethnicity, gender, sexual orientation, religion, age, physical or mental handicap, and others". Furthermore, Facebook[3] defined hate speech as "content that targets individuals on the basis of their real or perceived race, ethnicity, nationality, religion, sex, gender, sexual orientation, disability, or illness". Also, Twitter[4] defined hate speech as "Hateful conduct: You may not encourage violence against or directly attack or threaten other people because of their race, ethnicity, national origin, sexual orientation, gender, gender identity, religious affiliation, age, handicap, or serious disease".

One feature that we can observe and adapt in all of the definitions above is that hate speech is an attack on people's identities. Taking this into account, we developed a working definition of hate speech for this study to offer a common specific boundary for the Amharic hate speech labeling procedure. Hence, we define hate speech as *textual social media comments that promotes discrimination against individuals or groups based on their nationality, ethnic and religious affiliation, gender, or disability*.

2.2 Hate Speech Detection Approaches

Automatic hate speech detection research has increasingly relied on feature engineering techniques and classification algorithms. The effectiveness of the classification algorithms are heavily reliant on the feature engineering technique employed. The two most popular techniques are handcrafted feature engineering in machine learning and automated feature learning in deep learning.

Handcrafted Feature Engineering
Finding the right features to address a problem could be one of the most difficult challenges in machine learning, especially in hate speech detection. In hate speech detection, two types of features have been employed. On the one hand, there are general features that are used in text mining, and on the other hand, there are specific features that we only find in hate speech detection tasks and are intrinsically related to the characteristics of this problem, such as othering languages and stereotypes, which are not proposed in this work. Among the handcrafted feature engineering techniques which were used in several hate speech detection studies include dictionary-based [13], bag-of-words [14], N-gram [15] and [16], TF-IDF [15], Part-of-Speech [17], lexical syntactic features [18], rule-based [19], word sense disambiguation [20], topic modeling [2], and sentiment analysis [21]. Handcrafted feature engineering is vital, but it is time-consuming and error-prone. As a result, the scientific community proposed automated feature learning as an alternative approach.

Automatic Feature Learning
The massive volumes of data available on social media have provided tremendous opportunities for new knowledge discovery through the analysis of patterns of relations

[3] https://www.facebook.com/help/135402139904490.
[4] https://www.twitter.com.

that coexist in the data. Learning algorithms can figure out the optimum parameters to use to build the highest performing model. As a result, hate speech detection studies utilizing automated feature learning using deep learning models have demonstrated impressive results [9, 22] and [23].

2.3 Hate Speech Detection Using Deep Learning Approaches

Several researchers developed hate speech detection methods using deep learning approaches particularly CNN models. For example, [8] detected hate speech tweets posted in seven different languages using CNN and character level representation. In terms of accuracy, the highest results were 88.93% for a dataset with five languages and 83% for a dataset with seven languages. Furthermore, [9] used a CNN model with 300 dimensions and pre-trained word embeddings (GloVe and FastText) to detect hate speech against women and immigrants on Twitter in a bilingual environment, including English and Spanish. For English and Spanish, the suggested model received F1 scores of 0.488 and 0.696, respectively. In addition, [10] looked at char n-grams, word Term Frequency Inverse Document Frequency (TF-IDF) values, Bag of Words Vectors (BoWV) over Global Vectors for Word Representation (GloVe), and task-specific embeddings learnt using FastText, CNNs, and LSTMs. The authors obtained an F1-score improvement of 18% using the proposed models. Finally, [11] used the CNN model with word2vec embedding on the Twitter hate speech dataset and compared it to the baseline Logistic Regression with character n-gram. With an F1-score of 78.3%, the CNN model outperformed the Logistic Regression model with an F1-score of 73.9% in a 10-fold cross-validation test.

The above mentioned research provided intriguing techniques for hate speech detection. However, to the best of our knowledge, the integration of multiple channels of the CNN model to generate improved features from each channel for low-resourced language is not widely studied, notably the CNN model. As a result, the primary goal of this research is to investigate how shared features of the multi-channel CNN model on top of the word2vec word embedding layer perform in hate speech detection more efficiently than features generated from a single-channel CNN model on a limited dataset, as in the case of an under-resourced language like Amharic.

3 Dataset Construction

In this work, we build a new hate speech dataset from social media using Amharic language, Ethiopia's national language. We focused on Facebook, which is regarded by Ethiopian and Ethiopian origin diaspora bloggers, politicians, and academics as the most important platform for political and social conversation in Amharic, a phenomenon shared with other countries where Internet penetration is low and "Facebook" is the Internet for many users [23].

3.1 Data Selection and Annotation

Since there is no benchmark dataset for Amharic hate speech detection, we built our own using the Ethiopian Broadcasting Corporation (EBC) Facebook page[5] and some chosen individual Facebook pages[6] that publish hateful comments. We extracted selected comments/posts pertaining to race, religion, and ethnicity using the Facepager API, resulting in a set of 30,000 comments between April 15, 2019 and December 15, 2019. A total of 5,000 comments/posts (1,370 Hate and 3,630 Not-Hate) were chosen at random for annotation, while the remaining 25,000 were not. Three annotators (two candidate PhD. in Linguistics and one MSc. in Law) manually annotated the selected samples as "Hate" or "not-Hate" from which 2,000 (1,600 for training and 400 for testing) examples were chosen according to the label agreement. The average Cohen's Kappa agreement score[7] was 80%, indicating a good agreement. The Ethiopian government's hate speech and misinformation prevention and suppression proclamation[8], as well as our definition of hate speech and the hate speech characterization lists proposed in [24], were provided to the annotators. Accordingly, a speech is labeled as "Hate" when:

- "the speech targets a group or individual as a member of a group (ethnicity, race, religion)"
- "the speech content in the message expresses hatred"
- "the speech causes a harm"
- "the speaker intends harm or bad activity"
- "the speech incites bad actions"
- "the speech is either public and directed at a member of the group"
- "the context makes violent response possible" (Table 1).

Table 1. Sample data from the Amharic hate speech dataset on two classes

No.	Posts/Comments in Amharic	Posts/Comments in English	Label
1	ግም ነሽ ሰው አታቂም ጥንበ እሱ ስለሀገር እንጂ ስለዘር አላወራም ደደብ ነሽ	You are an idiot; he hasn't mentioned race.	Hate
2	ለሀገሩ ከብር ሲል እረፍትና እንቅልፍ ያጣውን መሪ ማድነቅ ብቻ ሳይሆን የእሱን ፈለግ በመከተል ልንደግፈው ይገባል	We must not only admire and respect the tireless and sleepless leader just for sake of his country's greatness, but also follow in his footsteps.	not-Hate
3	በኦሮሚያ በሚኖሩ ትግሬዎች ላይ እርምጃ እንወስዳለን በገጀራ አንገታቸውን እንቆርጣለን	In Oromia, we will take action against the Tigreans and kill them.	Hate

[5] https://www.facebook.com/EBCzena.

[6] https://www.facebook.com/604407519910492.

[7] https://www.statisticshowto.com/cohens-kappa-statistic/

[8] https://www.accessnow.org/cms/assets/uploads/2020/05/Hate-Speech-and-Disinformation-Prevention-and-Suppression-Proclamation.pdf.

3.2 Data Preprocessing

In this work, three typical data preprocessing procedures were completed: formatting, cleaning, and sampling. We created text and comma separated value (csv) files throughout the formatting process. We removed Amharic punctuation marks, URLs, unnecessary white spaces, and non-Amharic characters during the data cleaning process. In Amharic, a similar sound can be represented by many characters like Ge'ez, an ancient Ethiopian language. In Ge'ez, each form has its own meaning, but in Amharic there is no clear cut rule that indicates its purpose and usage [35]. Hence, since there are different ways of writing the same Amharic word using different characters/Fidel/, we performed character normalization using a normalization tool[9]. The tool can make a single word out of a range of letters that are used to make multiple versions of the same word, which is crucial for dimension reduction. For instance, the Amharic word ዓለም (world) can have multiple writing styles such as ኣለም (world), ዐለም (world) that can be all converted to አለም (world). However, we did not use stop word removal for dimension reduction in this work. Because, we found that it carries significant meaning in hate speech detection. For example, "ትግሬ ገዳይ ነው" ("Tigrie is killer"). The stop word ነው (is), plays a significant role in labeling the statement as hate speech. We can capture this concept using CNN with n-gram models (3-g).

3.3 Feature Engineering Methods

This step involves extracting key features from the raw text and numerically expressing the retrieved features. We used two distinct feature engineering approaches in this study: n-grams and automated feature learning using word2vec models.

N-gram Based Feature Selection. We utilized the n-gram model as features, feeding the TFIDF (term frequency-inverse document frequency) values to the SVM machine learning model. The occurrence of a word is predicted using an n-gram model based on the occurrence of its n−1 preceding word. In this experiment, we put the unigram (n = 1) language model to the test as a feature for the SVM classifier.

Word2vec Feature Learning. Given the large amount of textual data accessible, classification models in most resourceful languages (e.g., English) benefit from automated feature learning approaches such as word2vec models. To take advantage of such models, we utilize the continuous-bag-of-words (CBOW) word2vec [25] word embedding model to generate features for our hate speech detection task.

4 Classification Models

4.1 Support Vector Machine (SVM)

This is a training algorithm that optimizes the distance between training patterns and the decision boundary [26]. This is a well-known machine learning approach for

[9] https://abe2g.github.io/am-preprocess.html.

classification, regression, and other learning tasks [27]. The Support vector classification (SVC) kernel in LIBSVM [28] is a technique for two-class and multi-class classification. As a baseline, we utilized SVM with linear classifier and TF-IDF features in the experiment.

4.2 Convolutional Neural Network (CNN)

CNN models, which were originally developed for computer vision, have now been proved to be useful for NLP and have produced great results [29]. CNN is intended to learn features automatically and adaptively. CNN is comprised of three main building levels. These are convolution, pooling, and fully connected layers. While the first two, convolution and pooling, extract features, the third, fully connected maps the extracted features into final output such as classification [30, 31].

The model shown in Fig. 2 is a slight variant of the CNN architecture of [30] shown in Fig. 1. Let $x_i \in \mathbb{R}^k$ be the k-dimensional word vector corresponding to the i^{th} word in a sentence. A sentence of length n padded where necessary is presented as:

$$x_1 = x_1 \oplus x_2 \oplus \ldots \oplus x_n. \tag{1}$$

Where $\oplus$ is the concatenation operator, and the convolutional operational consists of a filter $w \in \mathbb{R}^{hk}$, which is applied to a window of h words to produce a new feature. For instance, feature c_i is generated from a window of words $x_{i:i+h-1}$ by:

$$c_i = f(w \,.\, x_{i:i+h-1} + b) \tag{2}$$

Where $b \in \mathbb{R}$ a bias is a term and f is a nonlinear function (e.g. rectifier or tanh). This is done for every time step of the input sequence $c\{x_{i:h}, x_{2:h+1}, \ldots, x_{n-h+1:n}\}$ to produce a feature map of:

$$c_i = [c_1, c_2, \ldots, c_{n-h+1}] \tag{3}$$

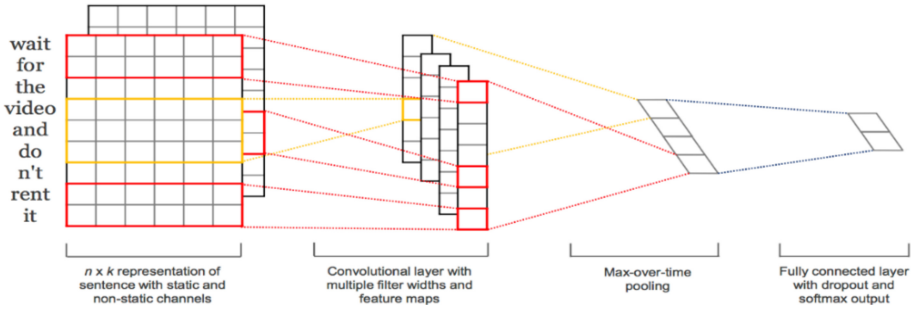

Fig. 1. CNN Architecture for natural language processing taken from [30].

5 Architecture of the Multi-channel CNN Model for Hate Speech Detection

We proposed a multi-channel CNN model for Amharic hate speech detection hoping that the multi-channel architecture would learn better features than the single-channel CNN model as shown in Fig. 2, especially for smaller datasets. The model involves using multiple versions of the standard model [30] with different sized kernels on the dataset. This allows the dataset to be processed at different widths of n-grams (groups of words) at a time, whilst the model learns how to best integrate these interpretations. After several experiments of n-grams (2, 3, 4, 5, 6, 7, 8) with multi-channels (2, 3, 4, 5, 6, 7, 8) we found better results with 4-g and 5-g with two channels. Hence, based on this experimental finding, we defined a multiple input model with two input channels for processing 4-g and 5-g. Each channel is comprised of the following elements:

- Input layer that defines the length of input sequences (Embedding layer set to the size of the vocabulary and 100 dimensional real valued representations).
- One-dimensional convolutional layer with 32 filters and a kernel size set to the number of words to read at once (4-g in one channel, and 5-g in different channel).
- Max-pooling layer to consolidate the output from the convolutional layer.
- Flatten layer to reduce the three dimensional output to two dimensional for concatenation.
- The output from the two channels are concatenated into a single vector and processed by a dense layer and an output layer.

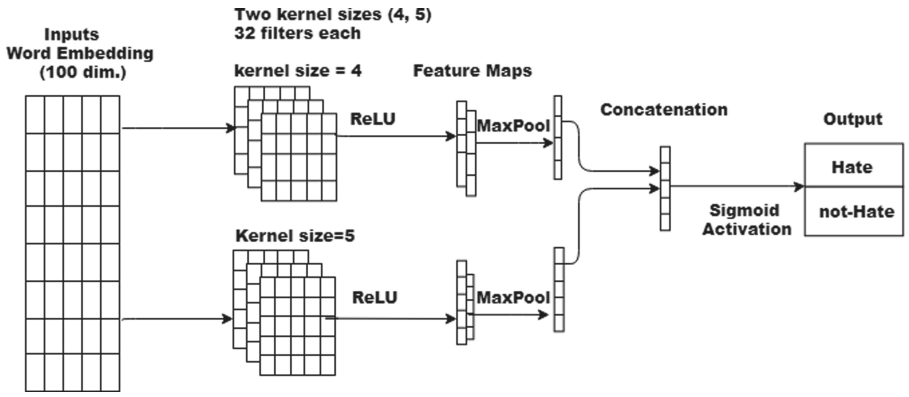

Fig. 2. Architecture of the proposed multi-channel CNN model for hate speech detection.

5.1 Major Component of the Amharic Hate Speech Detection Framework

The following key components make up the overall framework of the Amharic hate speech detection model, as illustrated in Fig. 3.

Data Source: It enable social media users post comments in Amharic, which is the source of data.

Facepager API: It helps collect comments posted by users on Facebook pages.

Preprocessing: It is to prepare textual data prior to training in order to get better classification results.

Word2vec/Input: It produces vector of words in the high dimensional space.

MC-CNN Model: It involves using multiple versions of the standard model with different sized kernels on the dataset as shown in Fig. 2.

Maxpool: It builds the embedding of a whole sentence from word representations. It takes the maximum value for each dimension of the word representations and builds a fixed-length vector by taking the maximum value for each dimension of the word representations. This yields a sentence representation in the same high-dimensional space as the word embedding.

Hate/Not-Hate: It is a differentiable classifier, which inputs the previously constructed sentence representation and outputs the final prediction, which is used to calculate the loss according to the ground truth, and to train the model.

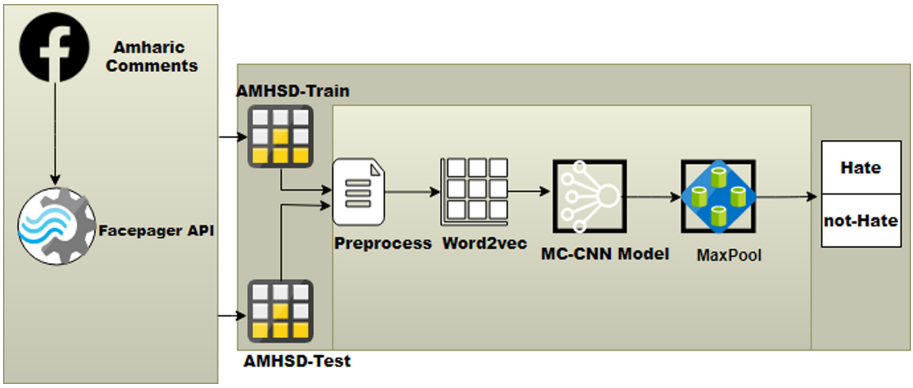

Fig. 3. The general framework of the Amharic hate speech detection (AMHSD) model.

6 Experiments

We did a series of experiments to test the proposed model for Amharic hate speech detection task (SVM, SC-CNN-1, SC-CNN-2, and MC-CNN). In all of these experiments, we used a binary classification task to classify social media comments as Hate speech or not-Hate speech. This section presents the description of the dataset utilized in the experiments, the experimental setups, baseline, and evaluation metrics.

6.1 Datasets

To train the proposed models for Amharic hate speech detection, we utilized the dataset mentioned earlier. The dataset includes 2,000 social media comments labeled as Hate or Not-Hate speech. We partitioned the data in an 80–20 ratio. That is, 1,600 (80%) of the data is used to train classification models to learn classification rules, while 400 (20%) is utilized to test the accuracy of classification models on new datasets.

6.2 Experimental Setups

We built the CNN models utilizing the Keras[10] framework and a TensorFlow[11] backend. The experiments are carried out on Google colab[12], which offers a free Jupiter note-book environment with GPU accelerator. For the CNN classifiers, we used word2vec features. The classifiers' particular settings are as follows:

Single-Channel-CNN-1 (SC-CNN-1): We built the SC-CNN-1 model using an embedding layer and one conv layer. The kernel has a size of 4 and the conv layer contains 32 filters. ReLu (Rectified Linear Unit) is the activation of the conv layer. The output layer is Dense 2 with sigmoid activation function, corresponding to two classes.

Single-Channel-CNN-2 (SC-CNN-2): The second SC-CNN-2 model is built with one embedding layer and one conv layer. The conv layer now contains 32 filters, and kernel size increased to 5 to accommodate more n-gram words. The activation of the conv layer is ReLu. Dense 2 with sigmoid activation function is the output layer, corresponds to two classes.

Multi-channel-CNN (MC-CNN): The MC-CNN model is built with an embedding layer and two conv layers. Each conv layer contains 32 filters, and the kernel sizes are 4 and 5 concatenated. The activation of the conv layer is Relu. Dense 2 with sigmoid activation function is the output layer, which corresponds to two classes.

6.3 Baseline

As a baseline, we used the Support Vector Machine (SVM) classifier, since it demonstrated effective classification performance in previous studies [32] employing keyword-based TFIDF feature engineering techniques. To build the classification model, we used Python's scikitlearn[13] library.

6.4 Model Evaluations

In this subsection, we have presented the assessment of the constructed classifiers to predict the classes of the unlabeled datasets as "Hate" or "not-Hate" speech using the

[10] https://keras.io/.

[11] https://www.tensorflow.org/.

[12] https://colab.research.google.com/notebooks/intro.ipynb.

[13] https://scikit-learn.org/stable/.

test dataset. The number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) generated by the classifiers are used to evaluate the performances of the models.

- True Positives (TP) are the number of correctly predicted Hate comments;
- True Negatives (TN) are the number of correctly predicted not-Hate comments;
- False Positives (FP) are the number of incorrectly predicted Hate comments;
- False Negatives (FN) are the number of incorrectly predicted not-Hate comments;

Furthermore, three performance metrics have been used to evaluate the classifiers: recall, precision, and F-measures [33].

Recall: is the proportion of actual positives which are predicted positive.

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

Precision: is positive predicted value. It is the proportion of predicted positives which are actually positive.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

F-measure: It is the harmonic mean of precision and recall.

$$\mathrm{F-measure} = 2 \cdot \frac{Recall\ .\ Precision}{Recall + Precision} \tag{6}$$

7 Results and Discussions

7.1 Results

In this section we present the experimental results. Table 2 shows the confusion matrix of all the models (SVM, SC-CNN-1, SC-CNN-2, and MC-CNN). As shown here, out of 200 actual Hate classes, SVM correctly classified 195 comments, SC-CNN-1 correctly classified 142 comments, SC-CNN-2 correctly classified 126 comments, and MC-CNN correctly classified 149 comments. However, SVM incorrectly classified 5 comments, SC-CNN-1 incorrectly classified58 comments, SC-CNN-2 incorrectly classified 74 comments and MC-CNN incorrectly classified 51 comments to the not-Hate class. On the other hand, in detecting the not-Hate class, the performance of the models were as follows. Out of 200 actual not-Hate test datasets, SVM correctly classified 175 comments, SC-CNN-1 correctly classified 186 comments, SC-CNN-2 correctly classified 183 comments, and MC-CNN correctly classified 177 comments. However, SVM incorrectly classified 25 comments, SC-CNN-1 incorrectly classified 14 comments, SC-CNN-2 incorrectly classified 17 comments, and MC-CNN incorrectly classified 23 comments.

Table 2. Confusion matrix of the four models.

	Predicted hate				Predicted not-hate			
	SVM	SC-CNN-1	SC-CNN-2	MC-CNN	SVM	SC-CNN-1	SC-CNN-2	MC-CNN
Actual hate	195	142	126	149	5	58	74	51
Actual not-hate	25	14	17	23	175	186	183	177
Total	220	156	143	172	180	244	257	228

Furthermore, the results of the comparative analysis of SVM, SC-CNN-1, SC-CNN-2, and MC-CNN using precision, recall and F1-score is shown in Table 3. The F1-score is used to evaluate the model performance. In Table 3, SVM and MC-CNN outperformed better than the other models in detecting the hate class. SVM has an F1-score of 92.8%, whereas MC-CNN has an F1-score of 80.2%. The SC-CNN-1 achieved an F1-score of 78.3%, whereas the SC-CNN-2 achieved an F1-score of 74.9%. Similarly, SVM and MC-CNN performed much better than other models in recognizing the not-Hate class, with F1-score detection accuracy of 92.1% and 82.7% respectively. Whereas the SC-CNN-1 performed with an F1-score of 80.0%, the SC-CNN-2 performed with an F1-score of 81.5%. As a result, when the four models were compared, the SVM outperformed than the other convolutional neural network models. When the performance of the convolutional neural network learning models were compared independently, the multi-channel CNN model outperformed than the other two single-channel CNN models.

Table 3. Evaluation of the four models.

Models	Hate			Not hate		
	P	R	F1	P	R	F1
SC-CNN-1	69.5	89.6	78.3	91.5	71.2	80.0
SC-CNN-2	63.5	91.4	74.9	94.0	72.0	81.5
MC-CNN	86.6	74.5	80.2	88.5	77.6	82.7
SVM	88.6	97.5	92.8	97.2	87.5	92.1

7.2 Discussions

In the last row of Table 3, for example, in determining the hate class, the SVM model has a precision of 88.6%, a recall of 97.5%, and an F1-score of 92.8%. These numbers provide important information about the model's classification accuracy when compared to the human annotator. The precision tells us the proportion of properly predicted hate speech comments (True positives) to the total number of predicted hate speech comments (True positives plus False positives). For example, a precision of 88.6% implies that the model can predict that 88.6% of all comments classified as hate speech by a human annotator are indeed hate speech comments. Normally, a high precision number suggests a low rate of false positives. A recall, on the other hand, shows us the proportion of properly predicted hate speech comments to the total number of comments in the actual hate speech class. For example, a recall of 97.5% from the SVM classifier indicates that out of the total number of real hate speech comments (200), the SVM model accurately predicts 97.5% of the hate speech classes. The F1-score, on the other hand, uses the average weights of accuracy and recall to generate a single score.

The experiments proved that the SVM classifier employing n-gram feature engineering techniques and TFIDF value outperformed than the CNN models. Nonetheless, when comparing single-channel CNN models to multi-channel CNN model, the multi-channel CNN model outperformed than the singlechannel CNN models. The theoretical analysis revealed that when comparing the four models, SVM outperformed than the other models in detecting both the Hate and not-Hate classes. Our initial assumption was that deep learning models in a multi-channel environment might surpass traditional techniques. However, this did not work effectively, as SVM with fewer datasets still outperformed the deep learning models. This is due to the fact that SVM is successful at classifying relatively small datasets with low training complexity. When the CNN versions SC-CNN-1, SC-CNN-2, and MC-CNN are compared independently, the MC-CNN outperformed the other single-channel CNN models. This can be due to the influence of shared features created by the mode's multiple channels with varied hyperparameter values. As a result, the proposed MC-CNN model can be viewed as a preferable alternative solution for hate speech detection in a deep learning settings where dataset scarcity is a concern as in the case of the Amharic language.

8 Conclusions and Future Works

In this study, we have presented an effective technique to classify Facebook comments written in the Amharic language as "Hate" or "not-Hate" speech by taking the advantages of the power of deep learning approaches. We have proposed a multi-channel CNN model based on the original CNN model for text classification on-top-of word2vec word embedding. The experiments were carried out using the Amharic hate speech detection dataset. The results of the experiments show the effectiveness of the proposed MC-CNN model compared to the SC-CNN model in a limited dataset. To the best of our knowledge, there is no prior work that used multi-channel features to detect hate speech for the Amharic language. Thus, this work is one additional contribution to the research undertaken for such under-resourced language. To apply the model for other languages, tuning the MC-CNN hyperparameters is fundamental. For the Amharic language, for instance, the model performs well at the 4-g and 5-g of words at two channels (other hyper-parameter settings being equal).

Though the experimental results are promising for a two-class hate speech detection and a monolingual small dataset scenario, further works can be done in the space of improving the model performance by properly considering the potential effect of relatively large datasets, multilingual datasets, and multiple hate speech classes such as hate speech in religion, ethnicity, gender, etc. Hence, a code-mixed dataset both from the resource-rich languages and under-resourced ethnic-based local languages can be tested to alleviate the fundamental problem of hate speech dataset scarcity and improve model performance. Finally, since we were not aware of the availability of a publicly hate speech dataset [35] at the time we didn't test our model on this dataset. Therefore, we will test the developed model on this particular Amharic hate speech dataset and a more standardized English hate speech dataset to test the generalization ability of the model in the next version of our work.

References

1. Plaza-del-Arco, F.M., Molina-González, M.D., Ureña-López, L.A., MartínValdivia, M.T.: Comparing pre-trained language models for Spanish hate speech detection. Exp. Syst. Appl. **166**, 114120 (2021)
2. Alshalan, R., Al-Khalifa, H., Alsaeed, D., Al-Baity, H., Alshalan, S.: Detection of hate speech in COVID-19-related tweets in the Arab region: deep learning and topic modeling approach. J. Med. Internet Res. **22**, e22609256 (2020). https://doi.org/10.2196/22609
3. Rawlence, B.: High stakes: political violence and the 2013 elections in Kenya. United States of America (2013)
4. Mossie, Z., Wang, J.H.: Vulnerable community identification using hate speech detection on social media. Inf. Process. Manage. **57**, 102087 (2020). https://doi.org/10.1016/j.ipm.2019.102087
5. Gagliardone, I., Pohjonen, M., et al.: Mechachal: online debates and elections in Ethiopia-from hate speech to engagement in social media. University of Oxford (2016)
6. Yimam, S.M., Ayele, A.A., Biemann, C.: Analysis of the Ethiopic Twitter dataset for abusive speech in Amharic. arXiv:1912.04419 (2019)
7. Elouali, A., Elberrichi, Z., Elouali, N.: Hate speech detection on multilingual twitter using convolutional neural networks. Rev. d'Intelligence. Artif. **34**, 81–88 (2020)
8. Ribeiro, A., Silva, N.: Convolutional Neural Networks for hate speech detection against women and immigrants on Twitter. Presented at the INF-HatEval at SemEval-2019 Task 5 (2019). https://doi.org/10.18653/v1/s19-2074
9. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: 26th International World Wide Web Conference, pp. 759–760 (2017). https://doi.org/10.1145/3041021.3054223
10. Gambäck, B., Sikdar, U.K.: Using convolutional neural networks to classify hate-speech. In: Proceedings of the 1st Workshop on Abusive Language Online, pp. 85–90 (2017). https://doi.org/10.18653/v1/w17-3013
11. Al-Hassan, A., Al-Dossari, H.: Detection of hate speech in social networks: a survey on multilingual corpus. J. Comput. Sci. Inf. Technol. **9**, 83–100 (2019). https://doi.org/10.5121/csit.2019.90208
12. William, M.C.: Hate speech | Britannica. https://www.britannica.com/topic/hate-speech. Accessed 26 Feb 2021
13. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: 25th International World Wide Web Conference, pp. 145–153 (2016). https://doi.org/10.1145/2872427.2883062
14. Pereira-Kohatsu, J.C., Quijano-Sánchez, L., Liberatore, F., Camacho-Collados, M.: Detecting and monitoring hate speech in twitter. Sensors (Switzerland) **19**, 4654 (2019). https://doi.org/10.3390/s19214654
15. Gaydhani, A., Doma, V., Kendre, S., Bhagwat, L.: Detecting hate speech and offensive language on twitter using machine learning: an N-gram and TFIDF based approach. CoRR abs/1809.0 (2018)
16. Waseem, Z., Hovy, D.: 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, San Diego California, USA, pp. 88–93 (2016). https://doi.org/10.18653/v1/n16-2013

17. Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., Bhamidipati, N.: Hate speech detection with comment embeddings. In: Proceedings of the 24th International Conference on World Wide Web, WWW 2015 Companion, pp. 29–30 (2015). https://doi.org/10.1145/2740908.2742760
18. Chen, Y., Zhou, Y., Zhu, S., Xu, H.: Detecting offensive language in social media to protect adolescent online safety. In: Proceedings of the 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust and 2012 ASE/IEEE International Conference on Social Computing, SocialCom/PASSAT 2012, pp. 71–80 (2012). https://doi.org/10.1109/SocialCom-PASSAT.2012.55
19. Haralambous, Y., Lenca, P.: Text classification using association rules, dependency pruning and hyperonymization. In: CEUR Workshop Proceedings, vol. 1202, pp. 65–80 (2014). https://doi.org/10.6084/m9.figshare.1189289.v1
20. Warner, W., Hirschberg, J.: Detecting hate speech on the world wide web. In: Proceedings of the 2nd Workshop on Language in Social Media, LSM 2012, pp. 19–26 (2012)
21. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Proceedings of the 11th International AAAI Conference on Web and Social Media, ICWSM 2017, pp. 512–515 (2017)
22. Fortuna, P., Bonavita, I., Nunes, S.: Merging datasets for hate speech classification in Italian. In: CEUR Workshop Proceedings, pp. 218–223 (2018). https://doi.org/10.4000/books.aaccademia.4752
23. Gagliardone, I., et al.: Mechachal: Online Debates and Elections in Ethiopia - From Hate Speech to Engagement in Social Media, 1 May 2016. https://doi.org/10.2139/ssrn.2831369
24. Fino, A.: Defining hate speech. J. Int. Crim. Justice **18**, 31–57 (2020). https://doi.org/10.1093/jicj/mqaa023
25. Mikolov, T., Le, Q. V., Sutskever, I.: Exploiting similarities among languages for machine translation. arXiv:abs/1309.4 (2013)
26. Boser, B.E., Guyon, I.M., Vapnik, V.N.: Training algorithm for optimal margin classifiers. In: Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, pp. 144–152 (1992). https://doi.org/10.1145/130385.130401
27. Cortes, C., Vapnik, V.: Support vector networks. Mach. Learn. **20**, 273–297 (1995)
28. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 1–27 (2011). https://doi.org/10.1145/1961189.1961199
29. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
30. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of Empirical Methods in Natural Language Processing, pp. 1746–1751 (2014). https://doi.org/10.3115/v1/d14-1181
31. Georgakopoulos, S.V., Tasoulis, S.K., Vrahatis, A.G., Plagianakos, V.P.: Convolutional neural networks for toxic comment classification. In: ACM International Conference Proceeding Series 2018, pp. 1–6 (2018). https://doi.org/10.1145/3200947.3208069
32. Kamble, S., Joshi, A.: Hate speech detection from code-mixed Hindi-English tweets using deep learning models (2018)
33. Seliya, N., Khoshgoftaar, T.M., Van Hulse, J.: A study on the relationships of classifier performance metrics. In: Proceedings of the International Conference on Tools with Artificial Intelligence, ICTAI, pp. 59–66 (2009). https://doi.org/10.1109/ICTAI.2009.25
34. Getachew, S.: Amharic Facebook dataset for hate speech detection. https://doi.org/10.17632/ymtmxx385m.1
35. Bender, M.L., Bowen, J.D., Cooper, R.L., Ferguson, C.A.: Language in Ethiopia. Oxford University Press, London (1976)