ITC 2/51
Information Technology and Control
Vol. 51 / No. 2 / 2022
pp. 356-375
DOI 10.5755/j01.itc.51.2.29988

Homophobic and Hate Speech Detection Using Multilingual-BERT Model on Turkish Social Media

Received 2021/10/18 | Accepted after revision 2022/02/22

http://dx.doi.org/10.5755/j01.itc.51.2.29988

HOW TO CITE: Karayiğit, H., Akdagli, A., Aci, C. İ. (2022). Homophobic and Hate Speech Detection Using Multilingual-BERT Model on Turkish Social Media. *Information Technology and Control*, 51(2), 356-375. http://dx.doi.org/10.5755/j01.itc.51.2.29988

Homophobic and Hate Speech Detection Using Multilingual-BERT Model on Turkish Social Media

Habibe Karayiğit, Ali Akdagli
Department of Electrical and Electronics Engineering, Mersin University, 33343, Turkey;
e-mails: d2014242@mersin.edu.tr, akdagli@mersin.edu.tr

Çiğdem İnan Aci
Department of Computer Engineering, Mersin University, 33343, Turkey; e-mail: caci@mersin.edu.tr

Corresponding author: d2014242@mersin.edu.tr

Homophobic expressions are a form of insulting the sexual orientation or personality of people. Severe psychological traumas may occur in people who are exposed to this type of communication. It is important to develop automatic classification systems based on language models to examine social media content and distinguish homophobic discourse. This study aims to present a pre-trained Multilingual Bidirectional Encoder Representations from Transformers (M-BERT) model that can successfully detect whether Turkish comments on social media contain homophobic or related hate comments (i.e., sexist, severe humiliation, and defecation expressions). Comments in the Homophobic-Abusive Turkish Comments (HATC) dataset were collected from Instagram to train the detection models. The HATC dataset was manually labeled at the sentence level and combined with the Abusive Turkish Comments (ATC) dataset that has developed in our previous study. The HATC dataset has been balanced using the resampling method and two forms of the dataset (i.e., resHATC and original HATC) were used in the experiments. Afterward, the M-BERT model was compared with DL-based models (i.e., Long-Short Term Memory, Bidirectional Long-Short Term Memory (BiLSTM), Gated Recurrent

Unit), Traditional Machine Learning (TML) classifiers (i.e., Support Vector Machine, Naive Bayes, Random Forest) and Ensemble Classifiers (i.e., Adaptive Boosting, eXtreme Gradient Boosting, Gradient Boosting) for the best model selection. The performance of the detection models was evaluated using F1-score, precision, and recall performance metrics. Results showed the best performance (homophobic F1-score: 82.64%, hateful F1-score: 91.75%, neutral F1-score: 96.08%, average F1-score: 90.15%) were achieved with the M-BERT model on the HATC dataset. The M-BERT detection model can increase the effectiveness of filters in detecting Turkish homophobic and related hate speech in social networks. It can be used to detect homophobic and related hate speech for different languages since the M-BERT model has multilingual pre-trained data.

KEYWORDS: Homophobic speech detection, multilingual BERT, transfer learning, deep learning, Turkish social media, sentiment analysis, text classification.

1. Introduction

Social media offers people a free platform to freely express their feelings. Users can share, disseminate their views, and write comments on other posts on social media [44]. There are constructive comments made to people on social media, as well as disturbing hate speech. Experiencing a large number of shares or interactions on social media every day and the decentralized structure of social media are among the most important reasons for the increase in hate speech [15, 11, 25]. Othering discourses encountered in society have been moved to these platforms with the frequent use of social media [17]. Othering with hate speech is a form of severe humiliation in terms of race, ethnicity, religion, gender, sexual orientation, disability, or disease [67]. Homophobic hate speech is sexual identity-based hate speech in which different sexual orientations are marginalized [31]. Homophobia, as a word, is a state of disdain and prejudice toward people with different sexual orientations for religious, social, and medical reasons [71]. People exposed to homophobic statements on social media are not always insulted because of their sexual orientation or behavior. For example, football players may be exposed to homophobic statements by their fans after losing matches [49]. Homophobic discourses are also used in the sense of cheating, being immoral, unreliable, perfidious, treacherous, vulgar, dishonest, characterless, and talkative.

Hate speech, which includes homophobic speech, is a behavior of discrimination, devaluation, and creating enemies. As a result, it leads to depersonalization, harassment, demeaning, intimidation, ignorance, and brutality of people or groups exposed to hate. Again, there are cases of silence and refusal to express themselves in people or groups exposed to hate. Depression and suicidality are other behaviors identified in individuals who are subject to hate speech [55]. Even if it is done on social media, it is necessary to control discourses before they turn into actions. Therefore, automatic language models should be developed to detect and prevent inappropriate content that is offensive to people [55].

The Instagram network was established on October 6, 2010, and the number of monthly active users worldwide is more than one billion. Worldwide, about two out of three people aged between 18 and 29 use Instagram [82]. 95 million shares are made daily on Instagram, and comments can be made on shared content [88]. Sentiment research for a certain purpose can be done and interpreted by collecting comments from Instagram. Social networks such as Instagram and Facebook delete comments that resemble hate speech in their databases to combat hate speech such as homophobia. Deleting or blocking comments does not mean that they are not a crime. The extent of insult is punishable, and it is mandatory to be followed by security forces. However, manual tracking is expensive and time-consuming. Developing a system that automatically detects and analyzes negative language is essential [51].

This study focuses on the detection of homophobic and related hate comments using the Homophobic-Abusive Turkish Comments (HATC) dataset [48]. The HATC dataset consists of 10,237 hateful, 1,226 homophobic, and 19,827 neutral Instagram comments that have been collected by the researchers. 256 of 1,226 homophobic comments were taken from the Abusive Turkish Comments (ATC) dataset which

has developed in our previous study [48]. The HATC dataset was balanced with the resampling method, and homophobic comments were determined by evaluating two forms of the dataset (i.e., HATC and resHATC) using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT) model, Deep Learning (DL) based classifiers (i.e., Long-Short Term Memory (LSTM), Gated Recurrent Unit (GRU) and Bidirectional Long-Short Term Memory (BiLSTM)), Traditional Machine Learning (TML) based classifiers (i.e., Naive Bayes (NB), Support Vector Machine (SVM), Random Forest (RF)), and Ensemble Classifiers (i.e. Adaptive Boosting (AdaBoost), eXtreme Gradient Boosting (XGBoost), and Gradient Boosting).

Contributions of this article can be summarized as follows:

1. A new Turkish homophobic dataset is presented [50].
2. There has been no previous study to distinguish homophobic comments in Turkish as far as we know. This is the first study in terms of both datasets obtained using homophobic data from Instagram and identification of Turkish homophobic comments by distinguishing them from multi-categories.
3. In addition to homophobic expressions, emojis related to homophobia were also taken into account in annotating the dataset.
4. The pre-trained M-BERT model achieved a very good F1-score than the other models in terms of all sentiment classes (i.e., homophobia, hateful, and neutral) values. The M-BERT model has the potential to be a suitable candidate for the homophobia detection model to be used in Turkish comment filters.
5. The M-BERT model used has pre-trained resources in 104 languages, and since it can take into account the format of different text languages, it can be used in studies of homophobic and hate speech in other languages.

The remainder of this article is organized as follows: Section 2 discusses previous work and current datasets on homophobic and related insults. Section 3 presents the materials and methods used in the study. Section 4 presents the experimental study and discusses the results. Finally, conclusions are given in Section 5.

2. Related Works

Studies on severe insult speech in social media were analyzed under different names and categories: hate speech analysis [4, 30, 19, 94], harassment detection [41, 35], abusive detection [49], aggression detection [20], misogyny detection [72], racism detection [56], flame detection [12], and offensive detection [97, 27]. Table 1 chronologically summarizes recent studies about hate speech regarding homophobia and sexual orientation on social media platforms.

Homophobic language analysis is generally classified together with other hate categories in studies conducted under headings of hate speech, offensive, and aggression. In a hate analysis study [79], hate expressions obtained from Twitter [91] and Whisper [95] have been classified into six hate categories (i.e., ethnicity, behavior, physical characteristics, sexual orientation, class, and gender). It is analyzed that the categories were similar on both social media platforms. In another study [27] for abusive language detection, tweets were labeled as homophobic and racist. Sexist expressions were labeled as offensive. In a study [29] conducted for offensive language detection in Portuguese, offensive data were classified as racism, sexism, homophobia, xenophobia, religious intolerance, and abuse. In a hate speech study in Italian [2], a dataset containing sexism, racism, and homophobic expressions was classified as homophobic or not homophobic. In a study [93] in which hate speech was categorized as ethnicity, religion, gender, or sexual orientation, hate speech was detected using feature templates. Also, racism, sexism, and homophobia categories were identified under the name of online hate speech using lexical and sentimental approaches. A method combination of dictionary-based algorithms and machine learning approaches was presented to predict hate speech under the categories of racism, sexism, homophobia in a dataset consisting of English tweets [93] In a study [8], authorship and aggression analysis have done for Mexican Spanish tweets in which the category of political humiliation, sexism, homophobia, and discrimination was defined as aggressive, and the other category was labeled as non-aggressive.

When we examine the source of the data used by the previous studies on homophobia, we see that most of the data were obtained from Twitter [94, 20, 75]. Data-

Table 1
Previous studies on detecting hate speech using homophobic categories

Paper Ref.	Lang.	Dataset	Category	Perf.
[93]	English-2012	Yahoo! and the American Jewish Congress (AJC) (1,000 paragraphs)	Race, Ethnicity, Gender, Sexual Orientation, Nationality, Religion, or Other Characteristic	0.63 F1-score
[79]	English-2016	Twitter, Whisper (20,305 tweets and 7,604 whispers)	Ethnicity, Behavior, Physical Characteristics, Sexual Orientation, Class or Gender	Not defined
[27]	English-2017	Twitter (24,802 tweets)	Racism, Sexism, Homophobia	0.90 F1-score
[29]	Brasilian Portuguese-2017	News website (10,336 comments posted for 115 news)	Racism, Sexism, Homophobia, Xenophobia, Religious Intolerance, Cursing	0.70 F1-score
[62]	English-2018	Twitter (975 tweets)	Racism, Sexism, Homophobia	80.56% Precision
[2]	Italian-2019	Twitter (1,859 tweets)	Homophobic, not Homophobic	0.80 F1-score
[70]	English, French, and Arabic-2019	Twitter (13,014 tweets)	Origin, Gender, Sexual Orientation, Religion, Disability, Other	0.86 Macro-F1
[8]	Spanish-2019	Twitter (10,856 tweets)	Politics, Sexism, Homophobia, Discrimination	0.65 Macro-F1

sets from Facebook [9], Instagram [49], YouTube [76], and other web platforms [97, 6, 28] are also available. When we examine the previous studies in terms of the methods used, Bag of Words (BoW), n-grams, DL-based (i.e., Convolutional Neural Network, Recurrent Neural Network (RNN), LSTM, GRU, and BiLSTM), and TML algorithms (i.e., Logistic Regression, NB, Decision Tree (DT), RF, and SVM) were frequently used in the detection of homophobia [94, 56, 27, 29, 2, 93, 63]. Due to the high classification success, DL-based algorithms have mostly been preferred for homophobia detection [33, 32, 10, 100]. In addition, pre-trained models based on transformer mechanisms have had significant classification successes in the analysis of hate speech [34, 101, 13].

Multilingual studies, which generally use TML and DL classification algorithms for hate speech detection, evaluate the robustness of proposed models in multiple languages simultaneously without experimenting in a cross-language environment [26, 70, 85]. The fuzzy logic method used in hate speech consists of logic categorizing values between 0 and 1. In most language problems, fuzzy logic algorithms are used to remove ambiguity and obtain precise classification results.

There are hate speech studies that used Fuzzy Rule-Based [38, 87], Fuzzy Multi-Task Learning [58], and Association Rule types [92].

3. Materials and Methods

This section presents the details of the datasets (i.e., HATC, and resHATC) used for the experiments and a summary of the classification algorithms.

3.1. The Homophobic-Abusive Turkish Comments (HATC) Dataset

The Turkish language belongs to the Altaic sub-division of the Ural-Altaic language family [54]. Turkic languages, consisting of 40 languages, are spoken as

a native language by almost 165-200M people in the world. Words with different meanings are obtained by adding morphemes such as "beads on a string" to a root word in the agglutinative Turkish language [68].

Turkish words can take many inflectional and derivational suffixes in a sentence. Expressions that change by taking a conjugation suffix in Turkish can correspond to a sentence in English.

gör+ebil+ecek+se+k → if we will be able to see

Figure 1 shows that the Turkish word "key" can take root five or more derivatives and end up as a modifier after five derivations.

Figure 1

Derivation process in a Turkish word

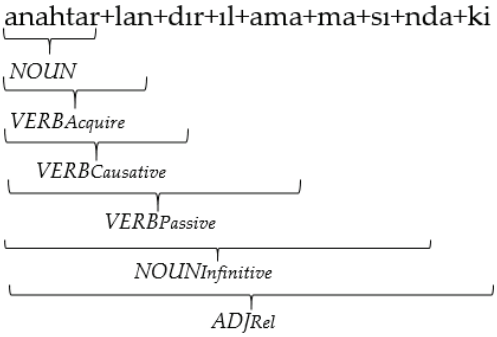


Table 2

Statistics about 18 frequently used Turkish words [68]

	Word	Morphemes	Ambiguity
1	bir	1	4
2	bu	1	2
3	da	1	1
4	için	1	4
5	de	1	2
6	çok	1	1
7	ile	1	2
8	en	1	1
9	daha	1	1
10	kadar	1	2
11	ama	1	3
12	gibi	1	1
13	var	1	2
14	ne	1	2
15	sonra	1	2
16	ise	1	2
17	o	1	2
18	ilk	1	1

Table 2 shows the 18 most frequently used words in a large Turkish corpus, along with the number of morphemes in the word and the morphological ambiguity for each. Most high-frequency words have relatively high morphological ambiguity, which corresponds to having different speech roots for words with one morpheme. In this study, a list of 201 words that would cause high morphological uncertainty was created and removed from the HATC dataset.

Datasets used in Natural Language Processing (NLP) studies are very important to improve classification performance. The HATC dataset consists of Instagram comments obtained from some accounts that have the potential to contain homophobic speech (i.e., @utandiran_paylasimlar, @kerimcandurmaz, @sametlicina) as well as the abusive Instagram comments in the ATC dataset which was developed in our previous study [48]. Abusive comments in the ATC dataset have sexist, homophobic, severe humiliation, and defecation expressions [49]. The comments in the ATC dataset were collected from accounts that are more likely to find hateful comments such as the Instagram accounts of the Turkish magazine page, football teams, and accounts of some football players. Table 3 shows the hateful Turkish words with the highest frequency in the ATC dataset.

Hate expressions in Turkish are usually root forms. In hate words that have a declension suffix, the meaning changes when stemming is done.

E.g; The word "şerefsiz (dishonest)" is hateful because it has a "siz" suffix. The root of the word "şerefsiz (dishonest)" is "şeref (honor)". The meaning of the word "şeref (honor)" is different than "şerefsiz (dishonest)" and does not contain hate. Therefore, the stemming process was not applied in the HATC dataset.

Homophobic comments were extracted from insult-labeled comments in the ATC dataset, combined with homophobic comments obtained from Instagram, and manually labeled as the homophobic cat-

Table 3
Hateful Turkish words with high frequency in ATC dataset

	Word	Type of hate	Frequency
1	siktir	Sexist	1182
2	amk	Sexist	917
3	orospu	Sexist	791
4	bok	Defecation Expressions	523
5	şerefsiz	Severe Humiliation	451
6	amına	Sexist	405
7	amik	Sexist	310
8	sikeyim	Sexist	298
9	aq	Sexist	288
10	piç	Severe Humiliation	260
11	sapık	Severe Humiliation	240
12	ananı	Sexist	207
13	top	Homophobic	185
14	mk	Sexist	159
15	yavşak	Severe Humiliation	156
16	mal	Severe Humiliation	146
17	kodumun	Sexist	131
18	ibne	Homophobic	108

egory. Sexist, severe humiliation, and defecation expressions in the ATC dataset were labeled as hateful comments. The remaining comments were labeled as neutral. Accordingly, 31,290 Instagram comments which 1,226 were homophobic, 10,237 were hateful, and 19,827 neutral were collected to form the HATC dataset (Table 4). Instagram Application Programming Interface [1] and Python programming language were used to gather homophobic data from Instagram. Instagram provides open-source unstructured data that allows for appropriate extraction and editing of data belonging to accounts accessible with a user account.

Labeling the homophobic dataset was carried out by two researchers according to the Big Slang Dictionary [3] regarding the Turkish Linguistic Society

Figure 2
Frequently used homophobic words in the HATC dataset

[89]. As shown in Figure 2, the most frequently used homophobic expression in Turkish is the word "top/ball" and its derivatives (i.e., topitoş, topitop, totoş, toplar/balls). Words such as "puşt, ibne/faggot", and "lavuk" are homophobic and are also used for unreliable and deceitful people. In addition, attention was paid to analyzing the comments that do not contain homophobic expressions but express homophobia with some emojis. The comments that have emojis such as ⚽, 🏐, 👌 were labeled as homophobic, other emojis were removed from the HATC dataset.

The HATC dataset is imbalanced as is seen in Table 4. There are many methods (i.e., data-level approaches and algorithm-level approaches) used to balance datasets: Data-level resampling techniques are used to normalize categorical distributions in imbalanced datasets. In resampling algorithms, the samples are reduced (i.e. under sampling) in the categories containing more samples in the training dataset, or the number of samples of the categories containing fewer samples in the training dataset is increased (i.e. oversampling) [90]. In our study, Random OverSampling

Table 4
Number of HATC dataset categories (i.e., homophobic, hateful, neutral)

Dataset	Number of homophobic comments	Number of hateful (i.e., sexist, severe humiliation, and defecation expressions)	Number of neutral comments
HATC	1,226 (3.9%)	10,237 (32.7%)	19,827 (63.4%)

(ROS) and Random UnderSampling (RUS) were utilized to balance the dataset. RUS consists of randomly removing examples of the majority class. The number of examples removed reduces the imbalance ratio, and it can balance the dataset, or even unbalance it in the opposite direction. ROS consists of randomly replicating examples of the minority class. As with the previous case, the number of examples generated reduces the imbalance ratio [46].

Classification performance can both improve and overfitting can be reduced on imbalanced datasets resampled using DL-based models. In oversampled networks, DL-based algorithms perform better, are more selective, learn faster, and the less it will over-fit [80].

In this study, the HATC dataset is divided into a training-test set using 10-fold cross-validation firstly (Figure 3). In each training dataset, the number of homophobic comments is randomly increased (oversampling) until it equals the number of hateful comments. At the same time, the number of neutral data is randomly reduced (undersampling) until it is equal to the number of hateful comments. Thus, the HATC dataset was balanced by resampling and the new dataset is called the resHATC dataset. Classification results of the HATC and the resHATC datasets were compared separately in the experiments. As shown in Figure 3, after dividing the HATC dataset with 10 cross folds, the number of Homophobic comments in the train set is 1,104, the number of hateful comments is 9,214, and the number of neutral comments is 17,845.

After applying the resampling technique, the number of homophobic comments is 9,214, the number of hateful comments is 9,214 and the number of neutral comments is 9,214. In the test set, the number of homophobic comments is 122, the number of hateful comments is 1,023 and the number of neutral comments is 1,982.

3.2. Methods

In this section, the algorithms utilized for the detection of homophobic and related hate comments are briefly presented. Methods of n-grams, Term Frequency - Inverse Document Frequency (TF-IDF), and Global Vectors (GloVe) were adopted for vectorized feature extraction. SVM, NB, and RF algorithms were used as TML algorithms. AdaBoost, XGBoost, and Gradient Boosting were employed as Ensemble Classifiers. LSTM, BiLSTM, and GRU methods were developed for DL-based classification.

The multilingual base model was used as the BERT model. Thus, a total of 22 combinations of resampling, feature subset selection, and classification models were trained and validated to classify homophobic comments. The schematic representation of the methods used in this study is given in Figure 4.

Figure 3

Balancing the number of the comments in the HATC dataset by resampling

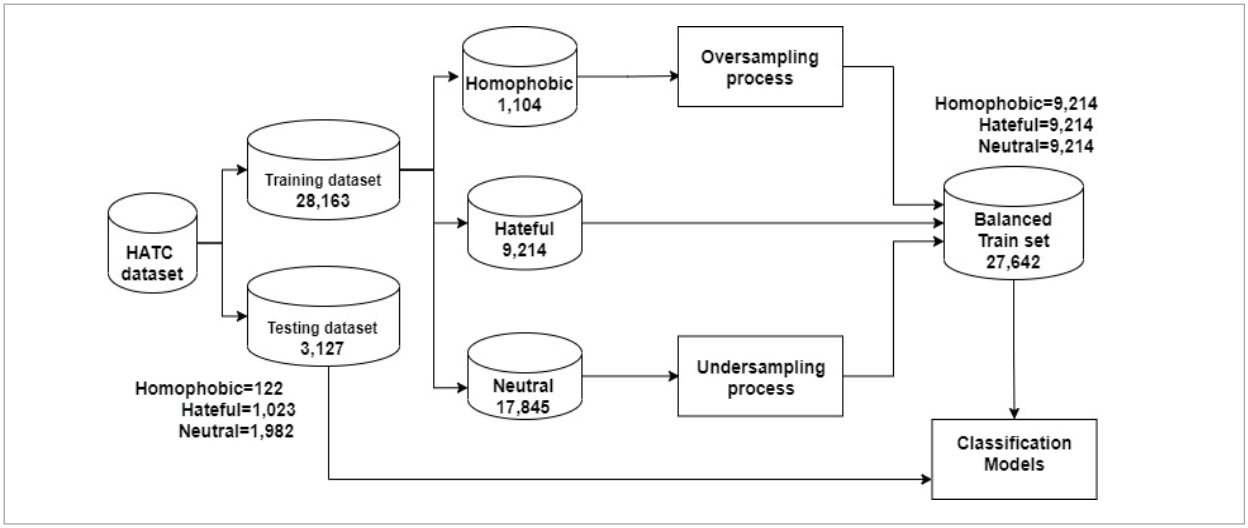
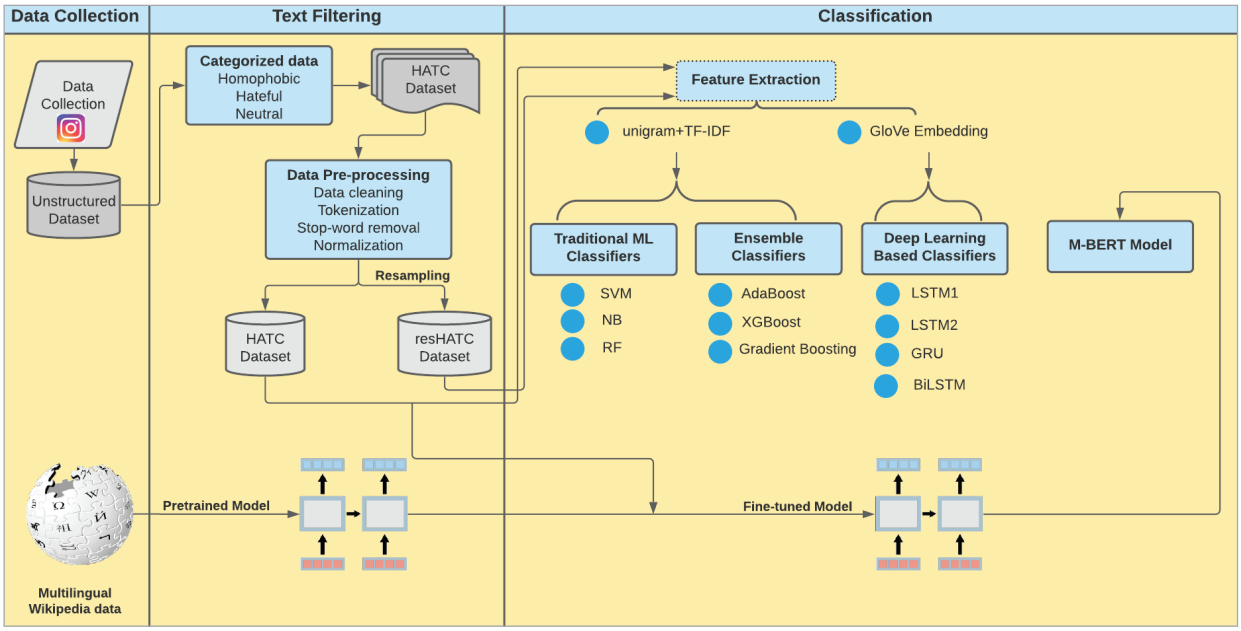


Figure 4
Homophobic and hate speech architecture



3.2.1. Preprocessing

In the preprocessing phase, the HATC dataset was cleaned by removing URLs, hashtags, numeric characters, punctuation marks, and emojis (except for homophobic ones) in comments. Comments in the dataset were separated into tokens and stop-words were removed.

3.2.2. Feature Extraction

Feature extraction is the stage of representing texts by converting them to numerical vectors [74]. n-gram feature representation creates a vocabulary of grouped words. Vocabulary consisting of single word structures is called the word-unigram model. The TF is the number of times a word occurs in a document while IDF is whether a word is common or rare across all documents [53]. TF-IDF feature extraction with word-unigram, which is a sparse vector representation, was used in this study for feature selection before applying TML and Ensemble Classifier models.

Word embeddings, which are numerical representations of words, aim to improve classification accuracy with a large number of pre-trained texts rather than training a small dataset to be used [18]. Word embedding algorithms carry semantic information while representing words and encoding the relationship between words [33]. In this study, the GloVe word embedding method, which creates word embeddings by collecting a global word-word co-occurrence matrix, was used with DL-based classifiers. The GloVe algorithm used in this study is trained on Common Crawl [24]. There are 253K words in the vocabulary and the dimension size is 300. Training data is web-crawled multilingual text with 2,736B tokens. The corpus size is 21 GB.

3.2.3. Traditional Machine Learning Models

The SVM classifier is a highly effective and well-known algorithm that can give successful results in text classification processes [39]. The SVM algorithm does not need a large amount of data to produce successful classification results. The purpose of the SVM algorithm is to find an optimal hyperplane for separating classes, and it is a classifier with solid theoretical foundations [77]. It reduces generalization error by an effective separation from both classes of hyperplane to the nearest training point [40]. The NB classifier is a simple classifier widely used in NLP problems such as hate speech and yields good results.

The principle of this classifier is based on Bayesian probability and assumes that probabilities of features are independent of each other. Assuming that all features are independent makes it easy to use feature selections such as BoW notation. The NB classifier is extremely fast in testing and estimation [98].

The RF classifier is essentially an ensemble learning approach. The RF algorithm is an advanced DT method that is frequently used in NLP studies. The DT algorithm has an unstable problem due to high variance. The RF classifier has been used to solve this problem. RF creates many different DTs, averaging scores obtained by DTs and it reduces bias with overfitting [16].

The grid-search algorithm is an algorithm that determines the most suitable parameters for a model by pre-classifying data [14]. Grid-search applies different parameter values within user-specified ranges to each model for the selection of the best combination of parameter values. In this study, parameter selection of the classification models was made by a grid-search technique using 10-fold cross-validation, and values for parameters of SVM were defined as follows; the cost parameter (C)={0.01, 0.1, 1, 10, 10.01, 10.1, 100, 100.01, 100.1} and kernel={rbf, linearSVC}. Testing small and large C values is a well-known approach in literature [7, 5] to get the best version of the SVM classifier. The scaling motivation behind the grid search process is carrying out a comprehensive evaluation of C parameters from soft (small C value) to hard (large C value) margins. The SVM model gave the best results with C=10.01 and kernel=linearSVC values. In the NB algorithm, the multinomial NB used for multi-class categories was chosen, and the Alpha value was determined as 0.1. For the RF algorithm, the n_estimators value was selected as 50. Optimal parameter values for all TML algorithms used are given in Table 5.

Table 5
Optimal parameters of TML classifiers

Classifier	Optimal Parameters
SVM	C=10.01, kernel=linear
NB	Alpha=0.1, MultinomialNB
RF	Number of estimators=50

3.2.4. Ensemble Models

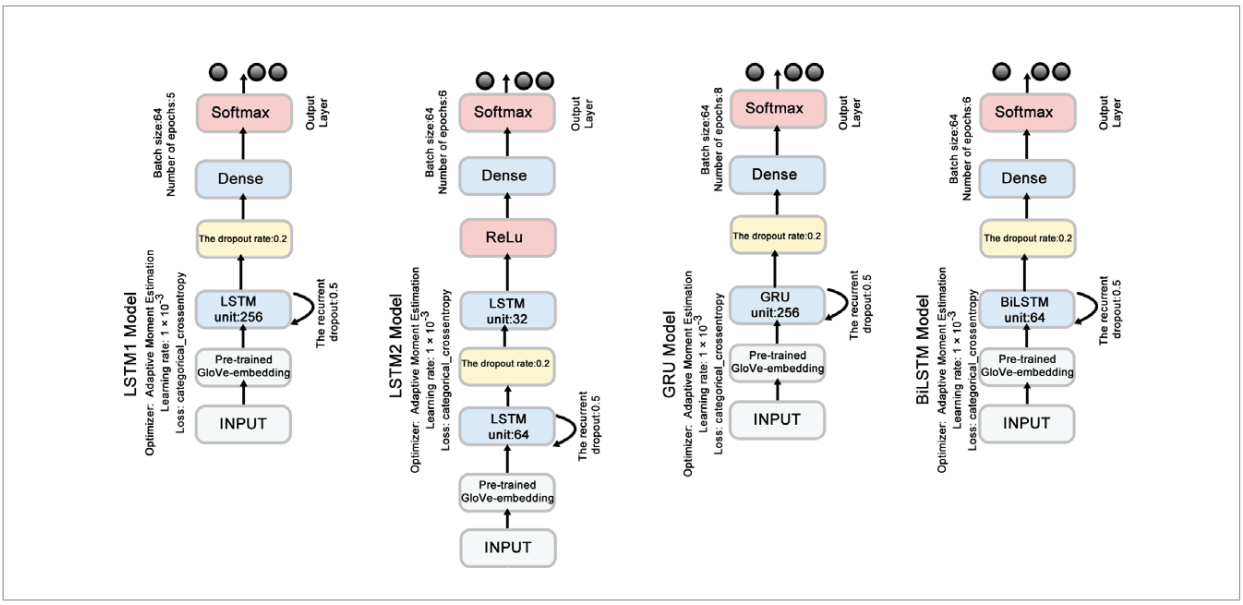
AdaBoost takes an iterative approach to building strong classifiers by learning from weak learner classifier errors. In the first step, DTs are used by AdaBoost as weak classifiers, and equally weighted values are given to the data. Weight values are updated according to results achieved in the first iteration [36]. AdaBoost thus reduces misclassifications [69]. Gradient boosting algorithms are effective classifiers for solving classification and regression problems that process data flexibly without the need for missing values. Overfitting and high variance in DTs are significantly reduced by gradient boosting utilizing a group of trees [66]. XGBoost is an ensemble learning method that applies a variant of gradient boosting based on DTs [21]. XGBoost combines several base DT learners to create a more robust model. Each base learner algorithm learns from the previous basic learner and reduces its error. As a result, the last learner has minimal bias and variance.

The tested parameter values for Ensemble Classifiers were defined as follows: n_estimators ={10, 20, 50, 100, 500,1000, 2000, 3000}. The optimal number of estimators' value was selected as 3000 by grid-search and it was applied to all Ensemble Classifier models.

3.2.5. Deep Learning Models

RNN is widely used in various tasks such as sequence classification, sequence labeling, and sequence generation [57]. RNN is a neural network in which the output of the previous step is fed as input data to the current step. The input data is processed according to the time series and the resulting output is utilized as the input for the next state [83]. Although RNN is durable in sequential modeling, it suffers from vanishing and exploding gradients in the long term. The LSTM algorithm was created to solve this problem through Forget, Input, and Output Gates. Forget Gate decides what to hide from prior steps. Input Gate decides what information to include after the current step, and Output Gate determines what will be the next hidden state [61]. GRU units similar to the LSTM algorithm are also used to solve vanishing gradient problems. The GRU algorithm has two gates, the Update Gate and a Reset Gate. The Update Gate acts similarly to an LSTM's forget and gate, it decides what information to keep and which to discard and what new information to add. The Reset Gate is used

Figure 5
Network architectures and optimal parameters of the DL-based classifiers



to decide how much of the previous information will be forgotten [22]. The BiLSTM unit that tries to capture text contexts consists of forwarding LSTM and backward LSTM units. This structure allows networks to have information about the sequence from two opposite directions at each step, both backward and forward [47].

In this study, parameter values for DL-based algorithms were determined by the trial-and-error method. 300-dimensional GloVe vectors were used to represent words in the dataset. Details of the network architectures and optimal parameters for all DL-based classifiers are given in Figure 5.

It has been tried by increasing the number of layers in DL models and the classification success has not decreased if only the LSTM model has two layers. Therefore, two models, one LSTM layered (LSTM1 model) and two LSTM layered (LSTM2 model), were created from the LSTM model. The network structure for the LSTM1 model is set as follows: LSTM layer (unit=256) - Dropout layer - Dense layer. The network structure for the LSTM2 model is set as follows: LSTM layer (unit=256) - Dropout layer – 2. LSTM layer (unit=256) - Dense layer. The network structure for the GRU model is set as follows: GRU layer (unit=256) - Dropout layer - Dense layer. The network structure for the BiLSTM model is set as follows: BiLSTM layer (unit=256) - Dropout layer - Dense layer. Dropout randomly removes entries between layers. Recurrent dropout eliminates entries between time steps. Dropout and recurrent dropout has a regularizing effect and can prevent overfitting. For all DL-based models, different dropout values (i.e. 0.2, 0.3, 0.4, and 0.5) were tried and the optimum dropout value was found as 0.2. Likewise, the optimum recurrent dropout value was used as 0.5. The Adaptive Moment Estimation (Adam) optimizer was used in DL-based models; the learning rate was 1×10^{-3}, and loss was categorical_crossentropy. During training, batch size is 64; number of epochs is LSTM1 model=5, LSTM2 model=6, GRU model=8, and BiLSTM model=6, respectively.

3.2.6. M-BERT Model

The BERT model is an unsupervised deep bidirectional neural network that implements bidirectional transformer architecture. A BERT-based transfer learning approach has started to be used frequently in hate classification studies, as it leads to increased classification performance and reduced training time [78]. The transfer learning approach also provides ef-

fective learning from limited labeled data with a pre-trained model. A pre-trained language model makes it easier to understand the current language even in data sources with few labels.

The BERT model logic is based on the attention mechanism, that is, the transformer structure, which learns the contextual relationships between words in a text. A basic transformer structure consists of an encoder that reads text inputs and a decoder that generates predictions for the task. The BERT model takes a sequence of fewer than 512 tokens as input data and gives a representation of the data as output. Tokenization is accomplished in two steps (the preliminary text normalization and punctuation splitting) with the WordPiece token [45]. The tokenized sequence is obtained by adding a [CLS] token at the beginning of each sentence and a [SEP] token at the end of each sentence. The BERT model performs text classification using the last hidden h state of the first token [CLS] as a representation of the resulting token sequences [81].

The M-BERT model is a pre-trained language model trained in the Wikipedia corpus of 104 languages [73]. The most important achievement of this model is that it is pre-trained on 104 different multilingual corpora and it performs quite well even in low-resource languages. In addition, the M-BERT model performs training taking into account the structures of all languages [37]. In this study, a pre-trained M-BERT model which supports 104 languages including Turkish with 12 stacked Transformer blocks, hidden dimensions 768, 12 self-attention heads, and overall 110,000,000 parameters was used. The M-BERT model used is capable of taking into account the format of different text languages by examining data from various languages [81].

In the BERT model used in our study, there are two dense layers with ReLU activation function, two dropout layers (0.2), and a dense layer with softmax activation function as the last layer. The BERT model was optimized using Adam optimizer and trained on a combination of BERT model with batch size (32), 3 epochs, and learning rate 1e-5.

3.2.7. Performance Metrics

A ten-fold cross-validation method was used for this study. While performing this process, the HATC dataset was divided into ten subsets, and each classification process was repeated ten times. Nine subsets were used as training datasets and one as a test dataset. An average result of ten folds was accepted as the final classification accuracy rate.

Choosing the optimum epoch number for training is another performance metric. When the epoch number is set high in the M-BERT and DL-based models, it may lead to overfitting and the training model may lose its generalization ability [42]. In recent years, techniques such as saving the best model or early stopping during training have been frequently used to reduce the risk of overfitting by the DL-based studies [64, 52, 84]. In our study, the early stopping technique was used to determine the epoch values most appropriately. After each epoch, the performance of the model was evaluated according to the Accuracy metric, and it was decided whether to stop the training or not. The training phase was finished when the increase in the Accuracy criterion stops or the maximum number of epochs allowed was reached. More specifically, the early stopping callback was used to stop training if the accuracy of the model did not improve more than 10 consecutive epochs. In our study, although we defined a training of 20 epochs as the initial parameter, the LSTM1 model stopped at the 5th epoch; the LSTM2 and BiLSTM models stopped at the 6th epoch; the GRU model stopped at the 8th epoch, and the M-BERT model stopped at the 3rd epoch. This approach contributes to avoiding overfitting in the models.

Precision, Recall, and F1-score were used to evaluate the performance of the proposed classification models as they are frequently used in hate speech analysis [27, 29, 2, 93, 62, 8, 70]. The confusion matrix summarizes the number of True and False samples predicted by the classifier [60]. True Negative (TN) is the number of (Actual) negatives that are correctly classified as negatives. False Negative (FN) is the number of (Actual) positives that are incorrectly classified as negatives. True Positive (TP) is the number of (Actual) positives that are correctly classified as positive. False Positive (FP) is the number of (Actual) negatives that are incorrectly classified as positives [99].

The Precision metric is the ratio of correctly classified positive samples (TP) to all samples classified as positive (TP+FP) (Equation (1)).

$$Precision = TP/(TP+FP). \quad (1)$$

The Recall metric is the ratio of correctly classified positives (TP) to all positive samples (TP+FN) in the dataset (Equation (2)).

$$Recall = TP/(TP+FN). \quad (2)$$

The F1-Score metric is found by the harmonic mean of the Precision and Recall metrics (Equation (3)).

$$F1\text{-}Score = 2*Recall*Precision/(Recall+Precision). \quad (3)$$

F1_macro averaging method was used in this study. Macro-averaged F1 provides a measured value for each label and calculates the average based on the number of labels in the dataset (Equation (4)).

$$Macro\_averaged\ F1 = \frac{1}{|Classes|}\sum_{i\in Classes} F1 - score(i) \quad (4)$$

4. Results and Discussions

The proposed classifiers were tested on the HATC and resHATC balanced datasets. All training and testing routines were performed on Google's free Colaboratory service [23]. Classification models consist of a feature extraction method and a classifier. Table 6 shows the performance metrics of the classification models with different combinations to detect homophobic expressions.

As seen in Table 6, the best F1-score is obtained from the M-BERT model in both datasets. The most important reason for that the transformer structure and attention mechanism can capture sentiment information better and more accurately. Using big data and vocabulary diversity in pre-trained different languages, the M-BERT model outperformed all approaches. The second-best model is BiLSTM in the resHATC dataset. Although LSTM1 and LSTM2 models alleviate gradient disappearance problems, the BiLSTM model was able to capture semantic information of context more effectively than LSTM models. The BiLSTM model helped learn bidirectional long-term dependencies between the forward-backward time directions and extracted better features from the LSTM models and the GRU model. The feature-enriched SVM model showed very close F1-score performance with the LSTM1, LSTM2, and GRU models in the HATC dataset. AdaBoost, XGBoost, and Gradient Boosting models gave better F1-score results in the HATC dataset than in the resHATC dataset. The resampling method had no effect on the TML and En-

Table 6
Performance comparison of the classification models for the homophobic category

Model	Homophobic Category		
	Precision (%)	Recall (%)	F1- score (%)
HATC+unigram+TF-IDF+ SVM	81.51	61.32	69.99
HATC+unigram +TF-IDF+ NB	96.52	33.40	49.63
HATC+unigram+TF-IDF+ RF	85.31	49.30	62.49
HATC+unigram+TF-IDF+ AdaBoost	59.32	47.03	52.46
HATC+unigram+TF-IDF+ XGBoost	81.93	53.94	65.05
HATC+unigram+TF-IDF+ Gradient Boosting	76.34	61.31	68.00
HATC+GloVe+ LSTM1	78.61	61.40	68.95
HATC+GloVe+ LSTM2	74.52	62.31	67.87
HATC+GloVe+ GRU	72.93	66.72	69.69
HATC+GloVe+ BiLSTM	75.62	67.52	71.34
HATC+M-BERT	90.81	76.29	**82.64**
resHATC+unigram+TF-IDF+ SVM	62.31	66.01	64.11
resHATC+unigram +TF-IDF+ NB	36.52	63.52	46.38
resHATC+unigram+TF-IDF+ RF	58.71	58.20	58.45
resHATC+unigram+TF-IDF+ AdaBoost	45.22	54.14	49.28
resHATC+unigram+TF-IDF+ XGBoost	50.73	67.83	58.05
resHATC+unigram+TF-IDF+ Gradient Boosting	56.22	67.52	61.35
resHATC+GloVe+ LSTM1	69.21	72.91	71.01
resHATC+GloVe+ LSTM2	69.21	68.51	68.86
resHATC+GloVe+ GRU	55.23	76.51	64.15
resHATC+GloVe+ BiLSTM	78.71	69.50	73.82
resHATC+ M-BERT	77.00	86.37	**80.88**

semble classifiers in terms of F1-score. Performance results of the classification models for the hateful category are given in Table 7. According to Table 7, the best model for the classification of hateful discourses is the M-BERT model in both datasets. The LSTM1 and LSTM2 models produced close F1 values to the second-best BiLSTM model in the HATC dataset. Table 8 demonstrates the classification models'

Table 7
Performance comparison of the classification models for hateful category

Model	Hateful Category		
	Precision (%)	Recall (%)	F1-score (%)
HATC+unigram+TF-IDF+ SVM	90.8	84.12	87.33
HATC+unigram +TF-IDF+ NB	85.4	86.23	85.81
HATC+unigram+TF-IDF+ RF	92.12	76.18	83.40
HATC+unigram+TF-IDF+ AdaBoost	86.31	74.61	80.03
HATC+unigram+TF-IDF+ XGBoost	94.9	80.22	86.94
HATC+unigram+TF-IDF+ Gradient Boosting	95.31	79.21	86.52
HATC+GloVe+ LSTM1	90.61	87.32	88.92
HATC+GloVe+ LSTM2	91.71	85.61	88.56
HATC+GloVe+ GRU	87.5	88.82	88.16
HATC+GloVe+ BiLSTM	89.01	88.84	88.92
HATC+M-BERT	94.02	89.65	**91.75**
resHATC+unigram+TF-IDF+ SVM	84.01	84.11	84.06
resHATC+unigram +TF-IDF+ NB	76.51	85.32	80.68
resHATC+unigram+TF-IDF+ RF	86.42	79.71	82.93
resHATC+unigram+TF-IDF+ AdaBoost	82.62	75.51	78.91
resHATC+unigram+TF-IDF+ XGBoost	90.22	78.91	84.19
resHATC+unigram+TF-IDF+ Gradient Boosting	90.81	79.1	84.55
resHATC+GloVe+ LSTM1	87.81	87.12	87.46
resHATC+GloVe+ LSTM2	84.82	87.11	85.95
resHATC+GloVe+ GRU	82.82	87.11	84.91
resHATC+GloVe+ BiLSTM	89.35	88.5	88.92
resHATC+ M-BERT	88.97	89.86	**89.06**

Table 8
Performance comparison of the classification models for neutral category

Model	Neutral Category		
	Precision (%)	Recall (%)	F1-score (%)
HATC+unigram+TF-IDF+ SVM	91.13	95.81	93.41
HATC+unigram +TF-IDF+ NB	91.21	95.22	93.17
HATC+unigram+TF-IDF+ RF	89.61	96.86	93.09
HATC+unigram+TF-IDF+ AdaBoost	87.42	94.68	90.91
HATC+unigram+TF-IDF+ XGBoost	88.63	98.01	93.08
HATC+unigram+TF-IDF+ Gradient Boosting	85.32	97.68	91.08
HATC+GloVe+ LSTM1	90.81	95.08	92.90
HATC+GloVe+ LSTM2	92.22	95.59	93.87
HATC+GloVe+ GRU	93.02	92.62	92.82
HATC+GloVe+ BiLSTM	93.51	94.61	94.06
HATC+M-BERT	94.56	97.67	**96.08**
resHATC+unigram+TF-IDF+ SVM	91.71	91.59	91.65
resHATC+unigram +TF-IDF+ NB	93.42	83.89	88.40
resHATC+unigram+TF-IDF+ RF	89.44	93.21	91.29
resHATC+unigram+TF-IDF+ AdaBoost	88.23	91.42	89.80
resHATC+unigram+TF-IDF+ XGBoost	89.52	93.40	91.42
resHATC+unigram+TF-IDF+ Gradient Boosting	89.61	94.41	91.95
resHATC+GloVe+ LSTM1	93.72	93.59	**93.65**
resHATC+GloVe+ LSTM2	93.71	92.62	93.16
resHATC+GloVe+ GRU	94.72	89.61	92.09
resHATC+GloVe+ BiLSTM	94.72	94.61	94.66
resHATC+ M-BERT	95.16	93.17	**93.99**

Table 9
Performance comparison for average three-class classification

Model	Average Performance		
	Precision (%)	Recall (%)	F1-score (%)
HATC+unigram+TF-IDF+ SVM	87.81	80.42	83.95
HATC+unigram +TF-IDF+ NB	**91.04**	71.62	80.17
HATC+unigram+TF-IDF+ RF	89.01	74.11	80.88
HATC+unigram+TF-IDF+ AdaBoost	77.68	72.11	74.79
HATC+unigram+TF-IDF+ XGBoost	88.49	77.39	82.57
HATC+unigram+TF-IDF+ Gradient Boosting	85.66	79.40	82.41
HATC+GloVe+ LSTM1	86.68	81.27	83.89
HATC+GloVe+ LSTM2	86.15	81.17	83.59
HATC+GloVe+ GRU	84.48	82.72	83.59
HATC+GloVe+ BiLSTM	86.05	83.66	84.84
HATC+M-BERT	93.13	87.87	**90.15**
resHATC+unigram+TF-IDF+ SVM	79.34	80.57	79.95
resHATC+unigram +TF-IDF+ NB	68.82	77.58	72.94
resHATC+unigram+TF-IDF+ RF	78.19	77.04	77.61
resHATC+unigram+TF-IDF+ AdaBoost	72.02	73.69	72.85
resHATC+unigram+TF-IDF+ XGBoost	76.82	80.05	78.40
resHATC+unigram+TF-IDF+ Gradient Boosting	78.88	80.34	79.60
resHATC+GloVe+ LSTM1	83.58	84.54	84.04
resHATC+GloVe+ LSTM2	82.58	82.75	82.66
resHATC+GloVe+ GRU	77.59	84.41	80.86
resHATC+GloVe+ BiLSTM	87.59	84.20	85.86
resHATC+ M-BERT	87.05	89.80	**87.98**

performance metrics for the neutral category. It was observed that models produced more successful F1-score values in determining the neutral category than detection of other categories (i.e., homophobic and hateful). The best model is the M-BERT model in the HATC dataset just like other categories' results.

The BiLSTM model produced the second-best F1 classification score in the resHATC dataset. Average performance results for the three categories (i.e., homophobic, hateful, and neutral) are presented in Table 9.

The overall performance comparison of the classification models is given below:

- Since the M-BERT model has the best classification performance (i.e., homophobic category F1-score: 82.64%, hateful category F1-score: 91.75%, neutral category F1-score: 96.08%) among all models used in the experiments, the average F1-score performance (i.e. 90.15%) is better than other models.
- The M-BERT model segments the space to better reflect the linguistic and evolutionary relationships between different languages in deep layers. It is aligned using dictionaries between languages, and cross-lingual embeddings can be learned collaboratively in completely unsupervised methods. The M-BERT model has been trained transfer learning between high-resource (70%) and low-resource (30%) languages with multilingual word embeddings and various levels of controls. In the M-BERT model, the Turkish language falls into the high-resource language group. It has been proven that the classification success of other languages with high source languages with the M-BERT model is close to the classification success of the Turkish language M-BERT model [96]. Therefore, the M-BERT model used in our experiments can be used for other languages and is recommended.
- The M-BERT model yielded higher F1-score performance values in the HATC dataset compared to the resHATC dataset in all categories. It is thought that the M-BERT model does not consider the problem of class imbalance, since it is a model with pre-trained sufficient Turkish data.
- When we consider the average performance of the three categories' F1-score results, the second-best model is the BiLSTM model in the resHATC dataset. The BiLSTM model, which processes data in both directions, may have performed better due to its ability to model sequential dependencies of a piece of text from both previous and consecutive contexts. The third best classification model is the LSTM1 model in the resHATC dataset.

- The BiLSTM model yielded higher F1-score performance values in the resHATC dataset compared to the HATC dataset in all categories. Although GloVe pre-trained word embedding is used as input sequences to the DL-based models, balancing the dataset has a positive impact on the classification success for the BiLSTM model.
- F1-score results of the SVM model in the HATC dataset are close to DL-based models' results. The performance of the SVM model in the resHATC dataset is worse than the results in the HATC dataset. The number of samples in each category does not affect the class boundary much, as the hyper-planes between the categories in the SVM algorithm are calculated according to the support vectors. Therefore, SVM is known to be potentially less susceptible to the class imbalance problem [86, 43]. However, it has been proven that the SVM algorithm gives good classification results on some resampling datasets [49, 65]. Balancing the dataset with resampling algorithms can give variable classification results (better or worse) in TML and Ensemble classifiers. Balancing the HATC dataset in this study decreased the F1-score performance of TML and Ensemble classifiers.
- The best classifier with the average F1-score result among Ensemble Classifiers is the XGBoost algorithm with 82.57% in the HATC dataset.
- The lowest average F1-score among all models was the NB classifier, with 80.17% in the HATC dataset. The NB classifier had the lowest classification result, with 72.94% in the resHATC dataset also.
- Adam optimizer is a substitute for stochastic gradient descent for training DL-based models. LazyAdam and AdamW methods were also evaluated in our study. LazyAdam is an upgraded version of Adam designed to be more efficient at handling sparse updates [84]. AdamW is a variation of Adam where the weight reduction is only performed after controlling the step size on a per-parameter basis [59]. However, using LazyAdam and AdamW optimization methods in our study did not affect the results. LazyAdam did not increase the classification results in DL-based models but caused a decrease in classification results compared to the Adam optimization in the M-BERT model. Besides, no improvement was observed in the performance of both models when the AdamW method was used instead of Adam.

5. Conclusions

In this study, the performance of the M-BERT, TML, DL-based, and Ensemble Classifier models was investigated to detect homophobic and related hate speech on Turkish social media. The architecture of the proposed detection system consists of data collection, preprocessing, feature extraction, and classification phases. First, a dataset related to homophobia was obtained from Instagram and combined with the ATC dataset. The dataset was used both in its original and balanced forms. It has been concluded that the M-BERT model is more successful than other models in classifying all categories (i.e., homophobic, hateful, neutral). In summary, it would be useful to use the M-BERT model in the detection of hate speech in Turkish. In future studies, multilingual classification success can be measured by using datasets in other languages. Different studies can be carried out by increasing data in the homophobic dataset and ensuring that the ATC dataset is divided into more categories (e.g., racism, sexism, severe humiliation, and defecation expressions).

References

1. Abu-Salih, B., Wongthongtham, P., Chan, K. Y., Zhu, D., CredSaT: Credibility Ranking of Users in Big Social Data Incorporating Semantic Analysis and Temporal Factor. Journal of Information Science, 2019, 45(2), 259-280. https://doi.org/10.1177/0165551518790424
2. Akhtar, S., Basile, V., Patti, V. A New Measure of Polarization in the Annotation of Hate Speech. In: International Conference of the Italian Association for Artificial Intelligence. Springer, Cham, 2019, 588-603. https://doi.org/10.1007/978-3-030-35166-3_41
3. Aktunç, H. Big slang dictionary of Turkish: (with witnesses). YapıKredi Yayınları, 2000.
4. Aljarah, I., et al. Intelligent Detection of Hate Speech in Arabic Social Network: A Machine Learning Approach. Journal of Information Science, 2021, 47(4), 483-501. https://doi.org/10.1177/0165551520917651

5. Almansour, N. A., et al. Neural Network and Support Vector Machine for the Prediction of Chronic Kidney Disease: A Comparative Study. Computers in Biology and Medicine, 2019, 109, 101-111. https://doi.org/10.1016/j.compbiomed.2019.04.017
6. Almerekhi, H., Kwak, H., Jansen, B. J., Salminen, J. Detecting Toxicity Triggers in Online Discussions. In: Proceedings of the 30th ACM conference on hypertext and social media, 2019, 291-292. https://doi.org/10.1145/3342220.3344933
7. Aoyagi, K., Wang, H., Sudo, H., Chiba, A. Simple Method to Construct Process Maps for Additive Manufacturing Using a Support Vector Machine. Additive manufacturing, 2019, 27, 353-362. https://doi.org/10.1016/j.addma.2019.03.013
8. Aragón, M. E., Carmona, M. A. A., Montes-y-Gómez, M., Escalante, H. J., Pineda, L. V., & Moctezuma, D. Overview of MEX-A3T at IberLEF 2019: Authorship and Aggressiveness Analysis in Mexican Spanish Tweets. In: IberLEF@ SEPLN. 2019, 478-494.
9. Aroyehun, S. T., Gelbukh, A. Aggression Detection in Social Media: Using Deep Neural Networks, Data Augmentation, and Pseudo Labeling. In: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), 2018, 90-97.
10. Badjatiya, P., Gupta, S., Gupta, M., Varma, V. Deep Learning for Hate Speech Detection in Tweets. In: Proceedings of the 26th international conference on World Wide Web Companion, 2017, 759-760. https://doi.org/10.1145/3041021.3054223
11. Banks, J. European Regulation of Cross-Border Hate Speech in Cyberspace: The Limits of Legislation. Eur. J. Crime Crim. L. & Crim. Just., 2011, 19, 1. https://doi.org/10.1163/157181711X553933
12. Bansal, A., et al. Classification of Flames in Computer Mediated Communications. arXiv preprint arXiv:1202.0617, 2012.
13. Benballa, M., Collet, S., Picot-Clemente, R. Saagie at Semeval-2019 task 5: From Universal Text Embeddings and Classical Features to Domain-Specific Text Classification. In: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, 469-475. https://doi.org/10.18653/v1/S19-2083
14. Bergstra, J., Bengio, Y. Random Search for Hyper-Parameter Optimization. Journal of Machine Learning Research, 2012, 13(2).
15. Bilge, R. The Construction of Hate Speech on Social Media and Legal Regulations on Hate Crimes. Yeni Medya, 2016, 1, 1-14.
16. Breiman, L. Random Forests. Machine Learning, 2001, 45(1), 5-32. https://doi.org/10.1023/A:1010933404324
17. Cammaerts, B. Radical Pluralism and Free Speech in Online Public Spaces: The Case of North Belgian Extreme Right Discourses. International Journal of Cultural Studies, 2009, 12(6), 555-575. https://doi.org/10.1177/1367877909342479
18. Catelli, R., Casola, V., De Pietro, G., Fujita, H., Esposito, M. Combining contextualized Word Representation and Sub-Document Level Analysis Through Bi-LSTM+ CRF Architecture for Clinical De-Identification. Knowledge-Based Systems, 2021, 213, 106649. https://doi.org/10.1016/j.knosys.2020.106649
19. Charitidis, P., Doropoulos, S., Vologiannidis, S., Papastergiou, I., Karakeva, S. Towards Countering Hate Speech Against Journalists on Social Media. Online Social Networks and Media, 2020, 17, 100071. https://doi.org/10.1016/j.osnem.2020.100071
20. Chatzakou, D., Leontiadis, I., Blackburn, J., Cristofaro, E. D., Stringhini, G., Vakali, A., Kourtellis, N. Detecting Cyberbullying and Cyberaggression in Social Media. ACM Transactions on the Web (TWEB), 2019, 13(3), 1-51. https://doi.org/10.1145/3343484
21. Chen, X., Yuan, Y., Orgun, M. A. Using Bayesian Networks with Hidden Variables for Identifying Trustworthy Users in Social Networks. Journal of Information Science, 2020, 46(5), 600-615. https://doi.org/10.1177/0165551519857590
22. Choe, D. E., Kim, H. C., Kim, M. H. Sequence-based Modeling of Deep Learning with LSTM and GRU Networks for Structural Damage Detection of Floating Offshore Wind Turbine Blades. Renewable Energy, 174, 218-235. https://doi.org/10.1016/j.renene.2021.04.025
23. «Colaboratory,» 2021. [Online]. Available: https://colab.research.google.com/. [Accessed: 05-Dec-2021].
24. «Common Crawl,» 2021. [Online]. Available: https://commoncrawl.org/2021/. [Accessed: 05-Jun-2021].
25. Çomu, T., Binark M. Hate Speech on Video Sharing Networks: Youtube Example. Doctoral dissertation, Ankara University, Institute of Social Sciences, Department of Women's, 2012.
26. Corazza, M., Menini, S., Cabrio, E., Tonelli, S., Villata, S. A Multilingual Evaluation for Online Hate Speech Detection. ACM Transactions on Internet Technology (TOIT), 2020, 20(2), 1-22. https://doi.org/10.1145/3377323
27. Davidson T., Warmsley D., Macy M., Weber I. Automated Hate Speech Detection and the Problem of Offensi-

ve Language. In: Proceedings of the International AAAI Conference on Web and Social Media. 2017, 512-515.

28. De Gibert, O., Perez, N., García-Pablos, A., Cuadros, M. Hate Speech Dataset from a White Supremacy Forum. arXiv preprint arXiv:1809.04444, 2018. https://doi.org/10.18653/v1/W18-5102
29. De Pelle, R. P., Moreira, V. P. Offensive Comments in the Brazilian Web: A Dataset And Baseline Results. An. do Brazilian Work. Soc. Netw. Anal. Min., 2017. https://doi.org/10.5753/brasnam.2017.3260
30. 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, 2015, 29-30. https://doi.org/10.1145/2740908.2742760
31. Dondurucu, Z. B. Sexual Identity Based Hate Speech in New Media: Inci Sozluk Sample. Gumushane Univ. E-Journal Fac. Commun., 2018, 6(2), 1376-1405. https://doi.org/10.19145/e-gifder.435744
32. Gambäck, B., Sikdar, U. K. Using Convolutional Neural Networks to Classify Hate-Speech. In: Proceedings of the first workshop on abusive language online, 2017, 85-90. https://doi.org/10.18653/v1/W17-3013
33. Gao L., Huang R. Detecting Online Hate Speech Using Context Aware Models. arXiv preprint arXiv:1710.07395, 2017. https://doi.org/10.26615/978-954-452-049-6_036
34. Gertner, A. S., Henderson, J., Merkhofer, E., Marsh, A., Wellner, B., Zarrella, G. MITRE at SemEval-2019 Task 5: Transfer Learning for Multilingual Hate Speech Detection. In: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, 453-459. https://doi.org/10.18653/v1/S19-2080
35. Golbeck, J., et al. A Large Labeled Corpus for Online Harassment Research. In: Proceedings of the 2017 ACM on Web Science Conference, 2017, 229-233. https://doi.org/10.1145/3091478.3091509
36. Gupta, N., Jindal, V., Bedi, P. LIO-IDS: Handling Class Imbalance Using LSTM and Improved One-vs-One Technique in Intrusion Detection System. Computer Networks, 2021, 192, 108076. https://doi.org/10.1016/j.comnet.2021.108076
37. Guven, Z. A. Comparison of BERT Models and Machine Learning Methods for Sentiment Analysis on Turkish Tweets. In: 2021 6th International Conference on Computer Science and Engineering (UBMK). IEEE, 2021, 98-101. https://doi.org/10.1109/UBMK52708.2021.9559014
38. Haque M. A. Sentiment Analysis by Using Fuzzy Logic. arXiv preprint arXiv:1403.3185, 2014. https://doi.org/10.5121/ijcseit.2014.4104
39. Hassan, Saeed-Ul, et al. Sentiment Analysis of Tweets Through Altmetrics: A Machine Learning Approach. Journal of Information Science, 2021, 47(6), 712-726. https://doi.org/10.1177/0165551520930917
40. Hemmatian F., Sohrabi M. K. A Survey on Classification Techniques for Opinion Mining and Sentiment Analysis. Artificial Intelligence Review, 2019, 52(3), 1495-1545. https://doi.org/10.1007/s10462-017-9599-6
41. Huang, B., Raisi, E. Weak Supervision and Machine Learning for Online Harassment Detection. In: Online Harassment. Springer, Cham, 2018, 5-28. https://doi.org/10.1007/978-3-319-78583-7_2
42. Ilias, L., Roussaki, I. Detecting Malicious Activity in Twitter Using Deep Learning Techniques. Applied Soft Computing, 2021, 107, 107360. https://doi.org/10.1016/j.asoc.2021.107360
43. Japkowicz, N., Stephen, S. The Class Imbalance Problem: A Systematic Study. Intelligent Data Analysis, 2002, 6(5), 429-449. https://doi.org/10.3233/IDA-2002-6504
44. Jenkins, J. A Sociolinguistically Based, Empirically Researched Pronunciation Syllabus for English as an International Language. Applied Linguistics, 2002, 23(1), 83-103. https://doi.org/10.1093/applin/23.1.83
45. Johnson, M. et al. Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. Transactions of the Association for Computational Linguistics, 2017, 5, 339-351. https://doi.org/10.1162/tacl_a_00065
46. Uez-Gil, M., Arnaiz-González, Á., Rodríguez, J. J., García-Osorio, C. Experimental Evaluation of Ensemble Classifiers for Imbalance in Big Data. Applied Soft Computing, 2021, 108, 107447. https://doi.org/10.1016/j.asoc.2021.107447
47. Kapočiūtė-Dzikienė, J., Tesfagergish, S. G. Senait Gebremichael. Part-of-Speech Tagging via Deep Neural Networks for Northern-Ethiopic Languages: POS Tagging via DNN for Northern-Ethiopic Languages. Information Technology and Control, 2020, 49(4), 482-494. https://doi.org/10.5755/j01.itc.49.4.26808
48. Karayiğit, H., İnan Acı, Ç., Akdagli, A. «Abusive Instagram Comments in Turkish | Datasets Novice | Kaggle,» 2020. [Online]. Available: https://www.kaggle.com/habibekarayiit/datasets. [Accessed: 08-Aug-2021].
49. Karayiğit, H., İnan Acı, Ç., Akdagli A. Detecting Abusive Instagram Comments in Turkish Using Convolutional Neural Network and Machine Learning Methods.

Expert Systems with Applications, 2021, 174, 114802. https://doi.org/10.1016/j.eswa.2021.114802

50. Karayiğit, H., İnan Acı, Ç., Akdagli A. https://www.kaggle.com/habibekarayiit/hatc-dataset (password: HATC). 2021.
51. Kasakowskij, T., Fürst, J., Fischer, J., Fietkiewicz, K. J. Network Enforcement as Denunciation Endorsement? A Critical Study on Legal Enforcement in Social Media. Telematics and Informatics, 2020, 46, 101317. https://doi.org/10.1016/j.tele.2019.101317
52. Kazi, M. K., Eljack, F., Mahdi, E. Predictive ANN Models for Varying Filler Content for Cotton Fiber/PVC Composites Based On Experimental Load Displacement Curves. Composite Structures, 2020, 254, 112885. https://doi.org/10.1016/j.compstruct.2020.112885
53. Khan, H. U., Nasir, S., Nasim, K., Shabbir, D., Mahmood, A. Twitter Trends: A Ranking Algorithm Analysis on Real Time Data. Expert Systems with Applications, 2021, 164, 113990. https://doi.org/10.1016/j.eswa.2020.113990
54. Kılınç, D., Özçift, A., Bozyigit, F., Yıldırım, P., Yücalar, F., Borandag, E. TTC-3600: A New Benchmark Dataset for Turkish Text Categorization. Journal of Information Science, 2017, 43(2), 174-185. https://doi.org/10.1177/0165551515620551
55. Kumar, A., Abirami, S., Trueman, T. E., Cambria, E. Comment Toxicity Detection via a Multichannel Convolutional Bidirectional Gated Recurrent Unit. Neurocomputing, 2021, 441, 272-278. https://doi.org/10.1016/j.neucom.2021.02.023
56. Kwok, I., Wang, Y. Locate the Hate: Detecting Tweets Against Blacks. In: Twenty-seventh AAAI Conference on Artificial Intelligence, 2013.
57. Liu, X., Qi, F. Research on Advertising Content Recognition Based on Convolutional Neural Network and Recurrent Neural Network. International Journal of Computational Science and Engineering, 2021, 24(4), 398-404. https://doi.org/10.1504/IJCSE.2021.117022
58. Liu, H., Burnap, P., Alorainy, W., Williams, M. L. Fuzzy Multi-Task Learning for Hate Speech Type Identification. In: The World Wide Web Conference, 2019, 3006-3012. https://doi.org/10.1145/3308558.3313546
59. Llugsi, R., El Yacoubi, S., Fontaine, A., Lupera, P. Comparison Between Adam, AdaMax and Adam W Optimizers to Implement a Weather Forecast Based on Neural Networks for the Andean City of Quito. In: 2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM). IEEE, 2021, 1-6. https://doi.org/10.1109/ETCM53643.2021.9590681
60. Luque, A., Carrasco, A., Martín, A., de Las Heras, A. The Impact of Class Imbalance in Classification Performance Metrics Based on the Binary Confusion Matrix. Pattern Recognition, 2019, 91, 216-231. https://doi.org/10.1016/j.patcog.2019.02.023
61. Mansoor, M., Ur Rehman, Z., Shaheen, M., Khan, M. A., Habib, M. Deep Learning Based Semantic Similarity Detection Using Text Data. Information Technology and Control, 2020, 49(4), 495-510. https://doi.org/10.5755/j01.itc.49.4.27118
62. Martins, R., Gomes, M., Almeida, J. J., Novais, P., Henriques, P. Hate Speech Classification in Social Media Using Emotional Analysis. In: 2018 7th Brazilian Conference on Intelligent Systems (BRACIS). IEEE, 2018, 61-66. https://doi.org/10.1109/BRACIS.2018.00019
63. Mehdad, Y., Tetreault, J. Do Characters Abuse More Than Words? In: Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2016, 299-303. https://doi.org/10.18653/v1/W16-3638
64. Montenegro, C., Santana, R., Lozano, J. A. Analysis of the Sensitivity of the End-Of-Turn Detection Task to Errors Generated by the Automatic Speech Recognition process. Engineering Applications of Artificial Intelligence, 2021, 100, 104189. https://doi.org/10.1016/j.engappai.2021.104189
65. Moraes, R., Valiati, J. F., Neto, W. P. G. Document-level Sentiment Classification: An Empirical Comparison Between SVM and ANN. Expert Systems with Applications, 2013, 40(2), 621-633. https://doi.org/10.1016/j.eswa.2012.07.059
66. Morris, C., Yang, J. J. A Machine Learning Model Pipeline for Detecting Wet Pavement Condition from Live Scenes of Traffic Cameras. Machine Learning with Applications, 2021, 5, 100070. https://doi.org/10.1016/j.mlwa.2021.100070
67. Mossie, Z., Wang, J. H. Vulnerable Community Identification Using Hate Speech Detection on Social Media. Information Processing & Management, 2020, 57(3), 102087. https://doi.org/10.1016/j.ipm.2019.102087
68. Oflazer, K., Saraçlar, M. Turkish and Its Challenges for Language and Speech Processing. In: Turkish Natural Language Processing. Springer, Cham, 2018, 1-19. https://doi.org/10.1007/978-3-319-90165-7_1
69. Onan, A. An Ensemble Scheme Based on Language Function Analysis and Feature Engineering

for Text Genre Classification. Journal of Information Science, 2018, 44(1), 28-47. https://doi.org/10.1177/0165551516677911

70. Ousidhoum, N., Lin, Z., Zhang, H., Song, Y., Yeung, D. Y. Multilingual and Multi-Aspect Hate Speech Analysis. arXiv preprint arXiv:1908.11049, 2019. https://doi.org/10.18653/v1/D19-1474
71. «Oxford Learner's Dictionaries | Find definitions, translations, and grammar explanations at Oxford Learner's Dictionaries,» 2021. [Online]. Available: https://www.oxfordlearnersdictionaries.com/. [Accessed: 09-Aug-2021].
72. Pamungkas, E. W., Basile, V., Patti, V. Misogyny Detection in Twitter: A Multilingual and Cross-Domain Study. Information Processing & Management, 2020, 57(6), 102360. https://doi.org/10.1016/j.ipm.2020.102360
73. Pires, T., Schlinger, E., Garrette, D. How Multilingual is Multilingual BERT? arXiv preprint arXiv:1906.01502, 2019. https://doi.org/10.18653/v1/P19-1493
74. Qi, X., Zhang, Y., Qi, J., Lu, H., Self-attention Guided Representation Learning for Image-Text Matching. Neurocomputing, 2021, 450, 143-155. https://doi.org/10.1016/j.neucom.2021.03.129
75. Sadiq, S., Mehmood, A., Ullah, S., Ahmad, M., Choi, G. S., On, B. W. Aggression Detection Through Deep Neural Model on Twitter. Future Generation Computer Systems, 2021, 114, 120-129. https://doi.org/10.1016/j.future.2020.07.050
76. Salminen, J., Almerekhi, H., Milenković, M., Jung, S. G., An, J., Kwak, H., Jansen, B. J. Anatomy of Online Hate: Developing a Taxonomy and Machine Learning Models for Identifying and Classifying Hate in Online News Media. In: Twelfth International AAAI Conference on Web and Social Media, 2018.
77. Saric, M., Dujmic, H., Russo, M. Scene Text Extraction in Ihls Color Space Using Support Vector Machine. Information Technology and Control, 2015, 44(1), 20-29. https://doi.org/10.5755/j01.itc.44.1.5757
78. Sharma, A., Kabra, A., Jain, M. (2022). Ceasing Hate with Moh: Hate Speech Detection in Hindi-English Code-Switched Language. Information Processing & Management, 2022, 59(1), 102760. https://doi.org/10.1016/j.ipm.2021.102760
79. Silva, L., Mondal, M., Correa, D., Benevenuto, F., Weber, I. Analyzing the Targets of Hate in Online Social Media. In: Tenth International AAAI Conference on Web and Social Media, 2016.
80. Simpson, A. J. Over-sampling in a Deep Neural Network. arXiv preprint arXiv:1502.03648, 2015.
81. Smetanin, S., Komarov, M. Deep Transfer Learning Baselines for Sentiment Analysis in Russian. Information Processing & Management, 2021, 58(3), 102484. https://doi.org/10.1016/j.ipm.2020.102484
82. «Social Media Use in 2021 | Pew Research Center,» 2021. [Online]. Available: https://www.pewresearch.org/internet/2021/04/07/social-media-use-in-2021/. [Accessed: 08-Aug-2021].
83. Song, T., Jiang, J., Li, W., Xu, D. A Deep Learning Method with Merged LSTM Neural Networks for SSHA Prediction. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2020, 13, 2853-2860. https://doi.org/10.1109/JSTARS.2020.2998461
84. Sowinski-Mydlarz, V., Li, J., Ouazzane, K., Vassilev, V. Threat Intelligence Using Machine Learning Packet Dissection. Transactions on Computational Science and Computational Intelligence, 2021.
85. Stappen, L., Brunn, F., Schuller, B. Cross-lingual zero-and few-shot hate speech detection utilising frozen transformer language models and AXEL. arXiv preprint arXiv:2004.13850, 2020.
86. Sun, Y. P., Wang, Y. F., Zheng, X. J. Analysis the Height of Water Conducted Zone of Coal Seam Roof Based on GA-SVR. Journal of China Coal Society, 2009, 34(12), 1610-1615.
87. Tashtoush, Y. M., Orabi, D. A. A. A. Tweets Emotion Prediction by Using Fuzzy Logic System. In: 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), IEEE, 2019, 83-90. https://doi.org/10.1109/SNAMS.2019.8931878
88. Thejas, G. S., Kumar, K., Iyengar, S. S., Badrinath, P., Sunitha N. R. AI-NLP Analytics: A Thorough Comparative Investigation on India-USA Universities Branding on the Trending Social Media Platform „Instagram". In: 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS). IEEE, 2019, 1-8. https://doi.org/10.1109/CSITSS47250.2019.9031050
89. «Turkish Language Institution Dictionaries,» 2021. [Online]. Available: https://sozluk.gov.tr/. [Accessed: 08-Aug-2021].
90. Tolba, M., Ouadfel, S., Meshoul, S. Hybrid Ensemble Approaches to Online Harassment Detection in Highly Imbalanced Data. Expert Systems with Applications, 2021, 175, 114751. https://doi.org/10.1016/j.eswa.2021.114751

91. «Twitter,» 2021. [Online]. Available: https://twitter.com/home?lang=tr. [Accessed: 06-Jun-2021].

92. Wadhwa, P., Bhatia, M. P. S. Classification of Radical Messages in Twitter Using Security Associations. Case Studies in Secure Computing: Achievements and Trends, 2014, 273-294.

93. Warner, W., Hirschberg, J. Detecting Hate Speech on the World Wide Web. In: Proceedings of the Second Workshop on Language in Social Media, 2012, 19-26.

94. Waseem, Z., Hovy, D. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In: Proceedings of the NAACL Student Research Workshop, 2016, 88-93. https://doi.org/10.18653/v1/N16-2013

95. «Whisper,» 2021. [Online]. Available: http://whisper.sh/. [Accessed: 06-Dec-2021].

96. Wu, S., Dredze, M. Are All Languages Created Equal in Multilingual BERT? arXiv preprint arXiv:2005.09093, 2020. https://doi.org/10.18653/v1/2020.repl4nlp-1.16

97. Wulczyn, E., Thain, N., Dixon, L. Ex Machina: Personal Attacks Seen At Scale. In: Proceedings of the 26th International Conference on World Wide Web, 2017, 1391-1399. https://doi.org/10.1145/3038912.3052591

98. Xiang, Z. L., Yu, X. R., Hui, A. W. M., Kang, D. K. Novel Naive Bayes based on Attribute Weighting in Kernel Density Estimation. In: 2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS). IEEE, 2014, 1439-1442. https://doi.org/10.1109/SCIS-ISIS.2014.7044787

99. Zeng, G. On the Confusion Matrix in Credit Scoring and its Analytical Properties. Communications in Statistics-Theory and Methods, 2020, 49(9), 2080-2093. https://doi.org/10.1080/03610926.2019.1568485

100. Zhang, Z., Luo, L. Hate Speech Detection: A Solved Problem? The Challenging Case of Long Tail on Twitter. Semantic Web, 2019, 10(5), 925-945. https://doi.org/10.3233/SW-180338

101. Zhang, H., Wojatzki, M., Horsmann, T., Zesch, T. ltl.uni-due at SemEval-2019 Task 5: Simple but Effective Lexico-Semantic Features for Detecting Hate Speech in Twitter. In: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, 441-446. https://doi.org/10.18653/v1/S19-2078