Psychosocial Features For Identifying Hate Speech In Social Media Text

ABSTRACT

This study uses natural language processing to identify hate speech in social media codeswitched text. It trains nine models and tests their predictiveness in recognizing hate speech in a 50k human-annotated dataset. The article proposes a novel hierarchical approach that leverages Latent Dirichlet Analysis to develop topic models that assist build a high-level Psychosocial feature set we call PDC. PDC organizes words into word families, which helps capture codeswitching during preprocessing for supervised learning models. Informed by the duplex theory of hate, the PDC features are based on a hate speech annotation framework. Frequency-based models employing the PDC feature on tweets from the 2012 and 2017 Kenyan presidential elections yielded an f-score of 83 percent (precision: 81 percent, recall: 85 percent) in recognizing hate speech. The study is notable because it publicly exposes a rich codeswitched dataset for comparative studies. Second, it describes how to create a novel PDC feature set to detect subtle types of hate speech hidden in codeswitched data that previous approaches could not detect.

Comment [u1]: Recheck this statement

Keywords: Hate Speech, Code-switching, Feature selection, Machine learning

1. INTRODUCTION

Hate speech is rhetoric that targets an individual or group based on protected characteristics like ethnicity, religion, or gender [1]. This needs a lot more attention than it is getting now. People of African and Asian heritage are being targeted with growing frequency in the US, as well as ethnic hatred and genocide in some African countries [3, 4, 5, 6]. Increasingly, campaign-related incidents trigger online public comments bordering on hate speech. Famously, politicians stir negative ethnic feelings, often provoking intense public reactions and counter-reactions on social media [5]. In Kenya, the lack of particular policy frameworks to hold media corporations, especially social media, accountable for hate speech promoted on their platforms exacerbates the situation. Instead, the legislation available during this study targeted individual users, including local administrators of social media networks like WhatsApp [6].

Social media user-generated content challenges traditional natural language processing, computational linguistics, and machine learning methodologies. In addition to being noisy and irregular, big data is also massive, diverse in data types, real-time created, and codeswitched. Codeswitching is a widespread social phenomenon that indicates group membership [7]. Although it is considered informal communication, codeswitching is becoming more prevalent in everyday communication among bilingual and multilingual populations, particularly on social media. Additionally, codeswitching on social media in regards to hate speech appears to be the de facto lingua franca for in-group membership. It is interpreted as enhancing cohesion in communicating with "our" people and separating oneself from others, particularly those perceived to be critical adversaries. Additionally, codeswitching is frequently used to emphasize an idea or item in communication. Given that some social media platforms allow users to interact anonymously, these platforms constitute fertile ground for the proliferation of hate speech.

Our work focuses on recognizing hate speech in codeswitched text messages received from social media. This is a difficult categorization issue that standard approaches have failed to address well, frequently by omitting the codeswitched text. An in-depth analysis of the data collected for this study indicated that some of the most pervasive hatred on social media is frequently disguised in coded text messages. Because social media communication is sometimes casual, it is not commonplace to encounter messages that alternate between various languages, particularly among multilingual communities. This complicates the task of parsing sentences and conducting contextual analysis on words and phrases using conventional monolingual methods. The shortage of resources for native languages, such as corpora, parts-of-speech taggers, and dictionaries. [8], coupled with undocumented grammatical rules and uncoordinated research networks, seem to exacerbate the situation [9]. As a result, extracting high-quality features from this type of data for machine learning applications requires a novel strategy that addresses the flaws in typical data processing techniques. As a result, the entire process of collecting, annotating, and selecting high-quality features that best describe hate speech in a codeswitching context for the goal of training a machine classifier becomes complex and expensive.

Previous research on hate speech identification has focused on monolingual datasets, with the most often used language being English. However, communication occurs on social media platforms in a variety of different regional and indigenous under-resourced languages, including Amharic, Bengali,

UNDER PEER REVIEW

Seneca, and Swahili. Given that more than half of the world's population is multilingual [10], we hypothesize that this number is increasingly reflected on social media through evidence of language code-switching. For example, practically the whole indigenous population in Kenya is multilingual, able to communicate in their mother tongue (L1), Swahili or Kiswahili, the national language (L2), and/or English, the official language (L2) [13].

However, whenever codeswitching occurs, whether at the word or sentence level, prior similar research regarded this content as noisy data and chose to delete the entire phrase during the preprocessing stage. Rather than dropping, is there any way to properly manage this increasingly popular social media phenomenon? Our study tries to address this challenge by examining a variety of variables to uncover distinguishing characteristics for training a computer classifier to identify hate speech in codeswitched text messages extracted from social media. As a result, our study fills a void in the field of automatic hate speech recognition for codeswitched language datasets. To our knowledge, this is the first effort to gather and train a classifier for a dataset of codeswitched languages, specifically English, Swahili, Sheng (slang), and a few instances of vocabulary from indigenous languages such as Gikuyu and Luo. As an illustration of a text message, consider the following:

"We will swear in the rightful president (RAO) on 12/12. Nyinyi Gikuyu mtabaki na uyo mwizi wenu. Raila won votes from all 39 tribes"[Translation of the Swahili codeswitched part: "You Kikuyus will be left with your thief"]

In this context, the purpose of our work was to investigate a methodology that more accurately captures essential characteristics of nuanced forms of hate speech, particularly in codeswitched text messages, to improve the effectiveness of machine categorization of large amounts of data. The primary objectives included the development of a conceptual framework for hate speech, the collection of a hate speech dataset from Kenyan social media, the training and evaluation of a hate speech classification model. This study makes two contributions. To begin, it creates and publicly releases a code-switched dataset of hate speech that may be used for comparative studies by other scholars. Second, the study proposes a novel psychosocial feature subset that captures language use in terms of psychosocial distancing, negative passion, commitment to hate, stereotyping, and hate as a story, to extract salient features that can be used to effectively train a machine classifier in identifying nuanced forms of hate speech in codeswitched text messages.

Comment [u2]: Plase list out the variables to uncover in this part with reference.

2. LITERATURE REVIEW

Earlier research has employed a variety of techniques to decipher the features of hate speech in text messages. Among these are the application of hate theories and frameworks. Critical race theory was utilized to develop standards for annotating a corpus of racist texts. [11]. Additionally, one study established a framework for analyzing offensive messages contained in text documents[12]. However, past research on critical race theory has been limited to classification based on race and the interaction of law and power. As a result, the theory falls short of defining other forms of hatred, such as those motivated by gender, religion, or disability. While some researchers have devised frameworks to assist in identifying harmful language[4, 15], these frameworks lack essential theoretical underpinnings and frequently rely on word lists. As a result, there is a need to close this gap, which this study does by developing a wholistic hate speech framework comprised of psychosocial characteristics and grounded in sound theory. This is meant to be sufficiently comprehensive to allow for the identification of different forms of hate in social media messages.

There is an increasing amount of research being conducted in the domain of hate speech, including automated methods for detecting hate speech[14, 15, 16] and other related topics such as offensive language identification[17, 18], cyberbullying [19, 20], radicalization and Terrorism [21, 22]. The studies on hate speech have handled the automatic classification problem in one of two ways: as a binary classification work or as a multi-class classification task. The former technique has been used in several earlier studies that focus on identifying subtypes of hate speech such as racism [15, 23, 24] and anti-Semitism [13]. It is not enough to recognize black and white in a multi-class classification test; it is also necessary to recognize the grey shade on the continuum of hate speech and non-hate speech (Ok) communications. Gray messages are recorded in this way by having an "offensive" class, which is analogous to how a human annotator would normally view and classify messages. Several previous studies have used multi-class categorization[26, 27, 28].

According to the review of these studies, numerous features have been deployed with varying degrees of effectiveness in increasing the detection of hate speech in text messages. These features can be divided into two categories: high-level features and low-level features. The high-level

UNDER PEER REVIEW

properties of a text message are human-readable and frequently qualitative notions that a human annotator may detect and use to classify the message. This category encompasses syntactic, stylistic, semantic, and lexical characteristics. The length of the message, the usage of part-of-speech tags, and the use of imperatives are all syntactic aspects. Stylistic characteristics include the usage of capital letters, exclamation points, emoticons, and character and punctuation overloading [11]. Semantic features include associational terms, hate verbs, negative polarity, and the use of subjective nouns. Lexical features include word lists containing accusatory and attributional keywords[19, 29], abusive words [28], insults or flames [17, 31, 32], and offensive language [18, 25, 33] that include racial slurs [24, 34].

Other common feature representations include Bag of words (BoWs), N-grams, and word embeddings. BoWs often result in a high recall value, [33] but low precision due to false positives. This is because the mere presence of hate or offensive terms in the message skews the classification towards the hate speech class without considering the context usage of the term [15, 25, 29]. The N-gram features can exist in two levels: as a character or word features. A key advantage of N-gram features is that they preserve context by keeping the word order in the original text. This feature has empirically shown better performance than BoWs in training machine classifiers [30, 36].

In contrast to the original text format, low-level features are often the extracted features that are amenable to machine processing, which means they may be used directly by a machine-learning algorithm to train a model. These are frequency counts computed using BoWs, N-grams, and word embedding representations such as count vectors, one-hot vector encodings, term frequency-inverse document frequency (TF-IDF), and dense vectors. In studies on hate speech identification, pre-trained word embeddings as dense vector representations are becoming increasingly popular as the preferred feature set for training deep learning algorithms [26, 37, 38]. Global Vectors (GloVe) [37], FastText n-grams, and Word2Vec text representations are all frequently used pre-trained embeddings at the character, word, and sentence levels. The frequency of use of both levels of features in prior hate speech research is summarised in Fig. 1.

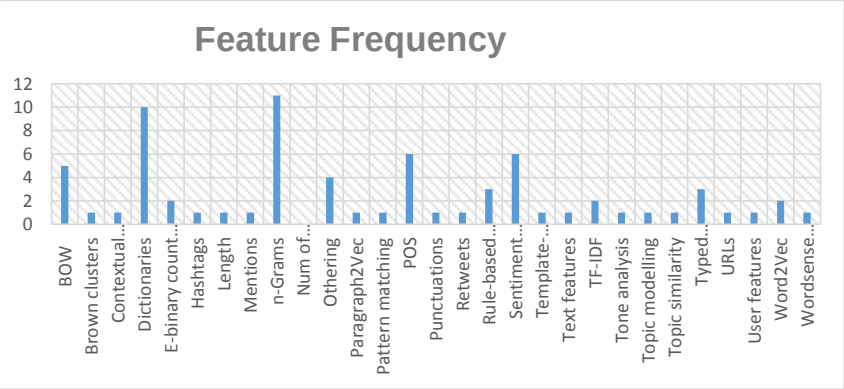

Fig. 1. Frequency of Features in hate speech studies

Notably, according to the reviewed literature, a large number of studies were conducted using English datasets, with just a small number of similar studies being conducted in other languages, such as Dutch [23], Amharic [38], Arabic [41, 42], but none in Swahili.

In general, the features that are employed for text classification play a critical part in defining the effectiveness and accuracy of the trained model when it comes to distinguishing between instances of different classes. It is necessary to identify the features, examine them, and select the most significant among them to inform the training of a machine classifier. Various characteristics have been used in past research to categorize hate speech. These, on the other hand, have frequently been convoluted, thus increasing the difficulty of comprehending and applying them. This research both theoretically and practically breaks down the complexity of these qualities into two basic groups, namely, high-level features and low-level features, to better understand them. Individuals who annotate high-level features report that they are easily comprehendible and immediately identifiable. As demonstrated in Fig. 2, these are further subdivided into psychosocial, linguistic, and App-specific features. As a result of this abstraction, a new methodology for capturing latent traits, such as the "othering" language, is introduced, which has proven useful in catching subtle kinds of hate speech in a prior study[41]. Furthermore, our study argues, using a comprehensive hate speech conceptual framework, that these

latent features are easily identifiable via psychosocial concepts and when combined, become informative features for identifying subtler forms of hate speech that conventional methods, particularly supervised machine learning, were insufficient to capture in the first place.

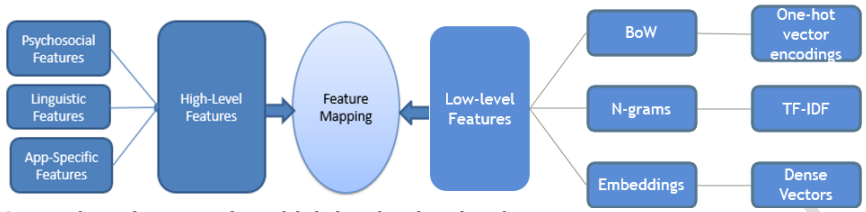

Fig.2 Mapping of Feature from high-level to low-level

3. METHODOLOGY

The four study objectives were addressed using a mixed-methods approach. To begin, a qualitative approach was utilized to define the discriminant characteristics of hate speech through an examination of important themes arising from diverse hate speech definitions and theories in the literature. These are succinctly expressed in fig.4's hate speech framework. Following that, the framework guided the collecting and manual annotation of tweets into three predetermined categories, namely Hate Speech, Offensive, or Neither. Following that, a quantitative approach was utilized to obtain word frequencies for each class, and other low-level features such as TF-IDF and word frequency vectors were employed to train the classifier model.

Using the Jupyter notebook integrated development environment, all operations from data pretreatment to data exploration and analysis, feature processing, model training, and actual classification were centralized. This was used to facilitate end-to-end model development and visualization of the data by utilizing Python programming (version 3.6.8) and machine learning libraries such as the natural language tool kit (NLTK) for data preprocessing, Pandas for visualizing and performing various operations on data, Scikit-learn for various types of machine learning models, and Matplotlib for data plotting, among others.

The study employed the ethnic group names of seven of Kenya's forty-two major tribes as the study population parameter [42], crawling tweets using the terms *Kikuyu, Luhya, Kalenjin, Luo, Kamba, Kisii, and Meru*, as well as their Swahili equivalents. Additionally, these ethnic names were employed in conjunction with other phrases to collect and create the raw dataset, as instructed by the multidimensional hate speech framework.

Unlike conventional research, which employs traditional sampling methods, big-data projects employ alternative sampling methods that computationally collect all available online content [43], such as by using a web crawler or Twitter API to collect a large number of messages from social media based on specific key words. Such methods are frequently free of the limits associated with classic sampling methodologies [44], which would have made it inefficient and impractical to collect a large volume of hate speech data from a large number of social media users in Kenya for machine learning purposes. Our study used simple random sampling to choose a study sample for annotation from the large volume of obtained data. In previous studies [25, 47], this sampling strategy was utilized to generate study samples from social media.

The CRoss-Industry Standard Processes for Data Mining (CRISP-DM) [46] was utilized to inform the five workflow procedures needed to accomplish the study's primary objective of establishing the salient elements required to develop a hate speech classifier for codeswitched communications. These steps comprised defining the problem, preparing data, processing features, modeling, and evaluating, as seen in Fig. 3.

UNDER PEER REVIEW

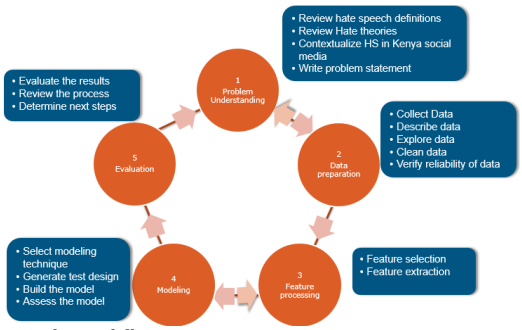

Fig. 3. The five-step research workflow

The experimental procedure was directed by these five phases, which have been shown to boost exploratory data analytics endeavors in the past[47], an approach that corresponded perfectly to the process activities incorporated in our study objectives. These are briefly covered in the subsections that follow.

3.1 Problem understanding

The purpose of this phase was to construct a working definition of hate speech for the study by first attempting to comprehend the environment in which the phenomena of hate speech exist to build a solution that is practicable within the problem's natural setting. In this regard, pertinent material, both online and in print, was rigorously researched to gain a thorough understanding of the problem of hate speech on social media in Kenya. The snowballing technique was employed to browse the cited articles that were referenced in the seminal literature research to gain further insights. Additionally, a qualitative analysis was conducted by examining the content of six hate theories, the definitions of hate speech contained within the user-content guidelines of various social media platforms, and the legal definitions of hate speech contained in Kenyan government policies, to identify relevant themes.

3.2 Data Preparation

This phase involved data collecting, data annotation, and data cleansing. Convenience sampling was used to collect tweets during the Kenyan presidential campaign in August 2017 and the runoff election in October 2017. Bootstrapping was the major data collection strategy. The strategy used to crawl Twitter for messages included the use of hate-related keywords [48], phrase patterns with a connotation of hate[49], offensive hashtags, and pro-hate user accounts [14]. Unlike other social media networks, messages published on Twitter are by default publicly available, topically structured, and programmatically accessible. Notably, several similar hate speech studies have used tweets[15, 28, 51]. Therefore, Twitter's API was used to build an application to collect tweets during the election week. A crawler built on python programming was also used to complement Twitter API's limitations of two weeks data collection window to acquire a formidable size of archived tweets dating as far back as the March 2013 Kenyan presidential elections. Besides, presidential campaign periods and events surrounding them are often prominent trigger events leading to spikes in online hate speech[51].

Human coders manually assigned a class to each message in the dataset. An initial team of forty undergraduate computer science students and staff members (80:20) was recruited and trained on the annotation scheme by convenience sampling. The group comprised of twenty-one male and nineteen female annotators, with an average age of twenty-three. The team's nationality was weighted towards Kenyans since annotators needed to understand the codeswitched nature of the corpus, which included messages in English, Swahili, and other native languages. The initial training used the annotation technique to help the team understand hate speech. The annotators were then instructed on how to annotate example messages through the study team's web-based annotation portal [9].

The original team of forty annotators was reduced to twenty-seven having dropped outliers. One week of annotating at least three thousand messages was required for selection. The first session provided valuable feedback on the annotation portal's speed. Previously, each tweet had to be annotated by a specific team, with one being a subject matter expert (SME). The revised design was inspired by the first session's sluggish annotation process and the necessity for a larger labeled dataset for machine classification training.

The second step was data cleaning, which aimed to improve the dataset's quality by removing noisy signals that could harm a machine learning model's training and overall performance. Empty rows,

UNDER PEER REVIEW

emojis, and punctuation were removed from the data. All words were lowercased to standardize the data. This was made possible by modules like Python's natural language toolkit (NKTK) and regular expressions libraries.

3.3 Feature Engineering

This phase's objective was to extract a subset of informative and high-quality vocabulary from the high-dimensional output in the data preparation stage. Following that, this textual subset comprised of high-level features needed to be translated into a low-level numeric representation amenable to machine learning via feature extraction. This is because machine learning algorithms are limited to processing numerical representations of features, such as vectors [52].

The high-level features were divided into two groups: the general vocabulary developed during the data processing step, and the PDC dictionary, which had five feature categories guided by the multidimensional hate speech framework seen in Fig. 4. Both categories were converted to BoW frequency counts, n-grams, and word embeddings. Following that, three low-level characteristics were retrieved from these: one-hot encodings, TF-IDFs, and dense vectors. The BoWs' features were determined based on the frequency of term occurrences in each communication. The n-grams were processed at the word and character levels, with n ranging from two to five. These were computed using the Scikit-learn machine-learning library's count vectorizer. The TF-IDF characteristics were used to compare the relevance of a particular term within a document and across the entire dataset. The overall goal is to penalize words that appear excessively frequently in all papers. This is because they may be less useful to the model than words that are unique to certain papers but are uncommon across all documents. As a result, the TF-IDF vectors for the various levels were created. Concerning Word Embeddings, the GloVe pre-trained embeddings were utilized to transform each word to a similar high-dimensional vector using the 100d file of around 1 million word vectors. To begin, the message dataset was tokenized. Following that, each token was mapped to its associated embeddings using the transfer learning approach.

The usefulness of these features was determined by training and comparing several classifiers. Besides, additional variables such as POS and topic models were extracted from the general lexicon and evaluated for their ability to improve the performance of classifiers.

Topic Models[53], were purposefully employed as high-level features for data exploration, associating the data with the conceptual framework, and, more crucially, as an automated procedure for determining which salient phrases to include in the subsequent step of creating the PDC word-family features. As a result, twenty-three semantically relevant topics or clusters were generated using the Latent Dirichlet Allocation (LDA) method from a vast corpus of short-text communications from social media. PDC traits are psycholinguistic characteristics derived from the triangle theory of hate's three dimensions of hate [54]. PDC espouses hate speech in three basic word families, all of which are concept-based and language-independent. As a result, the language list can be expanded or contracted by adding or eliminating terms with comparable meanings in different languages from their respective word families. The passion word family includes expressions of negative emotions including anger, fear, disgust, and disdain. These include threatening, abusive, disparaging, and other offensive language directed at a specific individual or group based on protected characteristics such as race, ethnic origin, or religion. An example message is "to *hell with all <group>. They need to be swept from this country*." Negative polarity and Sentiment analysis have been used in previous studies to detect passion instances [18, 33]. The **D**istance word family comprises terms that represent psychosocial distance or proximity in intergroup or interpersonal relationships, often known as "othering" language [50]. This is frequently suggested by a pronoun-heavy vocabulary [55, 56, 57, 58]. For example, *"us," them," they," we," you,"* etc. An example of an actual tweet is "*Kambas also do not make good leaders...they are Cowards*". The **C**ommitment word family is made of words or phrases that pledge to openly depreciate another person or group. This can be accomplished by referring to them as objects, insects, or animals, or by generalizing their inferiority, immaturity, or lack of humanity [59]. Additionally, this contains some of the code names that are only known to and used by the in-group to refer to out-group members. A sample tweet from our dataset is, "*Kikuyus Are Enemies of Luos. Stop Making Music with This Cockroaches*".

All of these high-level text attributes were encoded as input vector values for machine learning using the Scikit-learn toolkit. To be more precise, the text messages were converted to word count vectors using the CountVectorizer, and to word frequency vectors using the Tfidf Vectorizer. In all situations, the dataset's messages are tokenized first, and a vocabulary of known words is constructed. The output is an encoded vector containing the full vocabulary's length. Following that, each new text message is encoded as a fixed-length vector equal to the vocabulary's length. The CountVectorizer fills each vector location with a frequency count of each word occurrence in the new text message. If

a word in the new text message is not in the vocabulary, it is ignored and thus not counted in the resulting vector. The Tfidf Vectorizer evaluates word frequencies and assigns a high score to often occurring terms inside a document, but degrades the most frequently occurring words across all publications. The scores, which are frequently between 0 and 1, are used to weight the vector's frequencies when encoding new text messages.

3.4 Modeling

This procedure involved the selection of a model, training, and parameter tweaking. The classification models were trained using both conventional and deep learning strategies. A review of promising findings from prior similar investigations influenced the precise choice of machine learning techniques. The Naive Bayes, Support Vector Machine, Linear Logistic Regression, Decision Trees, and K-Nearest Neighbor were all examples of standard machine learning methods. Additionally, Bagging and Boosting models were utilized, namely Forest (RF) and Extreme Gradient Boosting (XGB). Convolutional Neural Networks and Hierarchical Attention Networks were used as deep learning techniques. Numerous machine learning experiments were conducted in this work using models equivalent to those found in Python's Scikit-learn machine learning model library.

When training each model, a set of hyperparameters was found and organized in a parameter grid. These were then automatically modified during the tests using Grid search and tenfold cross-validation[60] to score the combination of feature parameters and find the model's optimal hyperparameters. These included the soft margin cost, C, the kernel used in the estimation, and other estimator parameters. For the nonlinear Support Vector Machine, for example, the model's generalization in identifying various types of hate speech was evaluated by modifying the soft margin cost, C, to lower penalty values ranging from 0.001 to 1.0. Three popular kernels from the literature were used in the trials to assist the model in locating a nonlinear decision boundary, namely the linear, the Radial Basis Function (RBF), and the Polynomial. All of these model parameters were chosen from the SciKit-Learn libraries. Additionally, each time the algorithm was run, a pipeline was employed to smoothly merge these parameters with the vectorizer settings.

3.5 Evaluation

To mimic how our model might behave in the future, we partitioned the input dataset into training and testing sets. The confusion matrix was used to assess the trained models' accuracy performance by comparing predicted values to actual values from the test dataset. Additionally, the F-score was employed, which is based on the weighted average of precision and recall values. The highest prediction accuracy guided the selection of the optimal model for identifying the positive class, i.e., hate speech, with 10% serving as the validation data set, and model testing performed using K-fold (10-fold) cross-validation.

Based on the annotations made by a team of twenty-seven human annotators, an inter-rater reliability score was generated. At least three human annotators were required to annotate each tweet. Statistically, the mode decided the tweet's class, implying that the tweet's class was selected by two or more votes. In the event of a tie, a fourth annotator was introduced as a tie-breaker. Ideally, this fourth annotator was a subject matter expert. Because it could deal with missing values and outliers, Krippendorff's Alpha was chosen as an inter-rater reliability measure for the annotation exercise, which involved a team of twenty-seven novice annotators [61].

The triangulation approach was used to establish the construct and prediction validity of the research data and framework characteristics. This involved analyzing the performance of various traditional and deep learning machine learning techniques to select the optimal feature set for training our classifier.

3.6 Ethical Consideration

It is common for researchers to highlight concerns about user consent and user identity protection when using social media as the major source of data[62]. But unlike the privacy settings on other social networking sites, messages posted on Twitter are publicly available by default, unless the user chooses to activate the privacy settings. This is why our study focused on public tweets and retweets that do not require formal consent or ethical approval. Online users' identities were protected by replacing all user names and mentions with the generic label 'USERNAME'.

4. Results

The findings of the content analysis, data collecting, and processing, modeling, evaluation, and generalizability of the models are presented in this part.

4.1 Theoretical Framework for Hate Speech

UNDER PEER REVIEW

Distancing language, negative passion, devaluation, subjectivity, and stereotyping were identified as five key features of hate speech based on content analysis of many hate theories and hate speech definitions from various literature sources.

The use of pronouns in the text, particularly third-person plural nouns in English and Swahili, was indicative of distancing, also known as othering language. Several scholars have already utilized this idea to identify aspects of hate speech [43, 51, 56, 63]. Distancing is also obvious in social media messaging where one social group asserts superiority over another or isolates itself to safeguard the "purity" of the group membership. For example, during Kenya's post-election violence in 2007/2008, the Swahili phrase "madoadoa," which means "spots," was used to disseminate hate speech about non-natives by some politicians.

The negative passion dimension was marked by powerful feelings of hatred, fear, and antagonism toward the target individual or group. The material includes expletives such as swear words, obscenities, abusive, disparaging, and other offensive languages. This dimension has been used in several earlier research to identify hate speech [18, 33, 64]. Furthermore, negative passion was obvious in writing that incited violence against a person or a group because they shared a protected social feature. Devaluation is a hate-filled commitment characterized by the use of insulting language in text messages to refer to a target group as animals or insects. Using terms like maggots, cockroaches, rats, and so on to describe the target population. Other hate speech studies[4, 59] have employed this dimension.

Subjectivity was defined by the use of biassed or defective arguments, as well as the use of quantifiers and certainty phrases such as "always," "never," and "all." Stereotyping was defined as the use of a person's ethnic, racial, or religious group names to refer to them. Kikuyus, Luhyas, Kisiis, and other ethnic groups are examples. These five aspects, as well as their interactions, were included in the multidimensional hate speech framework, as depicted in Fig. 4.

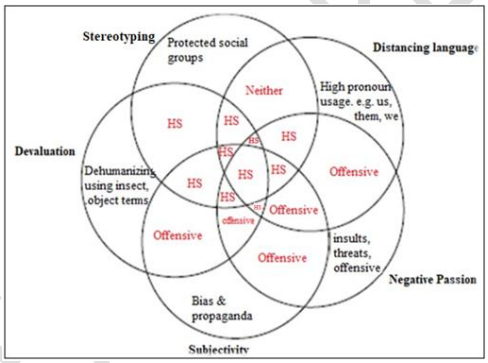

Fig. 4. The multidimensional Hate Speech Conceptual Framework

4.2 DATA

Around 400k unprocessed raw messages were gathered and stored in the comma-separated file (CSV) format. These mostly comprised of tweets from the August 2017 general elections in Kenya, as well as a runoff election held 60 days later in October 2017. Additional tweets from January to December 2017, as well as the March 2013 general elections, were crawled to create a sizable raw corpus.

The dataset included English, Swahili, and code-switched messages, with the majority of code-switched messages being English-Swahili. For instance

> "*yes I feel sorry for the dead people but* ***bado lazima tu wakikuyu wakae*** *like the guilty ones even when we are doing nothing.*"

A summary description of the dataset is shown in Table 1.

Table 1. Raw Dataset Description

Description	Number of text Messages
Total number of raw text messages collected	401,211
Total number of text messages after preprocessing	398,000
Codeswitched: Swahili, English, and others.	29309

60k messages were randomly selected for annotation by a team of twenty-seven human annotators from the 400k messages. Each tweet was annotated by a team of three annotators, with the class determined by majority vote. Around 50k tweets were annotated, with 6% containing hate speech, 19% containing offensive language, and 75% classified 'neither', as described in Table 2. The class of hate speech was in the minority. This is to be expected given the size of the social media dataset and is consistent with a prior study [68]. One of the conclusions of this study was that ethnic hate speech is the most prevalent form of hate speech in Kenya during election campaign periods. Thus, ethnic hate speech vocabulary could be used extensively and broadly as a domain for developing a classifier model for the Kenyan setting. Second, in contrast to binary classification systems, the addition of the "offensive" class helped distinguish clearly between hate and offensive communications, lowering the likelihood of mislabeling tweets as hate speech, a common error during annotation exercises [24].

Table 2. Class distribution of annotations

Class	Description	Count
0	Hate Speech	3094
1	Offensive	9401
2	Neither	37819
Total		**50314**

Using Krippendorf's Alpha, the inter-coder reliability score was 0.5207. This meant that the annotators were not in complete agreement half of the time. This is in line with prior studies[65] that had a lower inter-rater score of 0.17. The low inter-rater agreement has been attributed to a variety of personal sensitivities and societal prejudices, as well as the use of inexperienced but inexpensive annotators [66]. In our example, a few annotators who did not consistently attend the complete annotation course injected some teacher-noise into the annotations, lowering the score. Despite the training, the teacher noise, together with the team's tacit knowledge and biases during annotation, might form part of the latent qualities that were modeled as random components in the noise signal. Another reason for a large number of missing annotations could be that Krippendorff's Alpha assumes that each message is annotated by the entire team of annotators, in this case twenty-seven, whereas the annotation portal was designed to have each message annotated by a random team of three annotators from the twenty-seven. The necessity to maximize the volume of comments from a team of human coders while using the fewest resources possible inspired this approach. Random undersampling was used on the majority class, i.e. the 'neither' class, as well as the 'offensive' class, to better train a robust and unskewed classifier. This yielded a dataset of 9726k tweets that was fairly balanced, with majority votes in each of the three classes. In addition, a finer and more balanced dataset of 2537k tweets with just full agreement annotations was created. Both datasets were used to train machine learning algorithms in the studies and are freely available on Kaggle.

The Latent Dirichlet Allocation (LDA) model was utilized to discover deep underlying notions of hate in a large corpus of code-switched text using topic modeling [53]. LDA, a hierarchical probabilistic model, has previously been used to successfully identify cyberbullying-related subjects[17]. Each word in the corpus is represented by LDA as a finite mixture of underlying Passion, Distancing, and Commitment (PDC) subjects, which are modeled over an unlimited number of topics characteristic of a text document[53]. This aids in the development of a probabilistic model for the codeswitched corpus, which will give high probabilities to messages that are strongly related to the corpus' membership and other messages that are comparable to them. As a result, the LDA technique proved useful in data preprocessing and as a first-level statistical approach in automatically detecting and extracting passion, distancing, and discriminative (PDC) features from the huge corpus in this work. These characteristics were found in the twenty-three latent subjects recovered as a "bag of words" that were closely related to the hate speech category. The use of LDA, on the other hand, revealed the limits of the bag-of-words technique, which does not keep word order and hence does not preserve word meaning or context.

4.3 Modeling

The purpose of this work was to determine the predictive power of the psychosocial (PDC) feature set as compared to the traditional high-level features when training a classifier for hate speech. The conventional features included the lexical features (LEX) that defined the input corpus's generic lexicon. The BoWs and n-grams were retrieved, as well as Part of Speech (POS) and Application-specific (APP) variables such as the frequency of retweets and likes. As a result, nine machine learning models were trained and their performance compared to determine the best model for recognizing nuanced types of hate speech in codeswitched messages. To determine the optimal

UNDER PEER REVIEW

features and model performance, the features were tested singly and in combination using a feature combo. The wrapper technique was adopted, in which the PDC feature set began with only a few elements classified as psychosocial, as determined by the LIWC psychological word list[67]. The categories elements were expanded over time when additional features were uncovered in previously reported hate speech texts, as well as translation equivalents to account for codeswitched occurrences. Additionally, the Lex features were lacking in comparison to the comprehensive and instructive PDC elements. This can be explained again by the vectorizer's random feature sampling approach for extracting the Lex features from the input dataset, which includes a parameter for specifying the number of features. With text, as the number of features increases, the computation becomes more complicated, particularly in terms of memory and compute time necessary to handle the highly sparse input vector. In general, unlike the often wide and "diluted" Lex feature set, the PDC feature set consists of fewer but highly informative characteristics, i.e., "concentrated features," to identify hate speech. As illustrated in Fig. 5, supplementing the usual Lex feature set with PDC always resulted in improved performance. On the other hand, the inclusion of Lex or other features resulted in a decrease in performance. This is due to the noise created by these features and, consequently, the sparseness of the new input vector in classifiers.

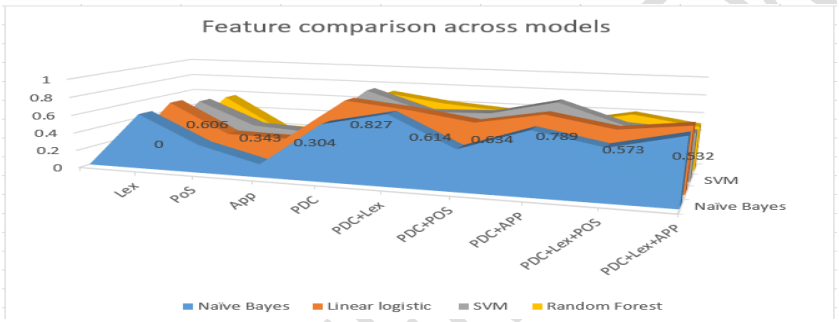

Fig. 5. Feature Comparison across models

Previous research on hate speech identification has relied on lexical and other natural language processing (NLP) elements. However, these kinds of capabilities will fall short of successfully capturing hate speech in codeswitched messages on their own. As a result, classifier models that explicitly use these traditional features will underperform, producing a high number of false negatives, in contrast to how hate is expressed in social media postings.

Psychosocial features (PDC) from the study, as well as other high-level features such as linguistic features (PoS), general lexical features (n-grams), and application-specific features (App), such as the length of a tweet, were used to train classifiers, and their performance was compared to that of other conventional text classification algorithms such as Nave Bayes, Linear Logistic Regression, Support Vector Machine, K-Nearest Neighbor, Decision Trees, Random Forest The models were trained using tenfold cross-validation on a dataset with an 80/20 ratio of training to testing characteristics. The models and feature categories with the highest accuracy performance across the seven machine learning techniques were compared and evaluated using a grid search algorithm. The Support Vector Machine model achieved the best accuracy of 76.2 percent, followed closely by the Linear Logistic Regression model, which achieved 75.8 percent accuracy. The accuracy scores for each model were determined using a tenfold cross-validation procedure and are well represented by the box and whisker plot in Fig. 6.

Given that the major purpose was to identify hate speech, the emphasis moved to the models' performance in identifying hate speech. As a result, just the accuracy performance of the two promising models was recovered. The experimental outcomes are described in Table 3 based on the balanced dataset.

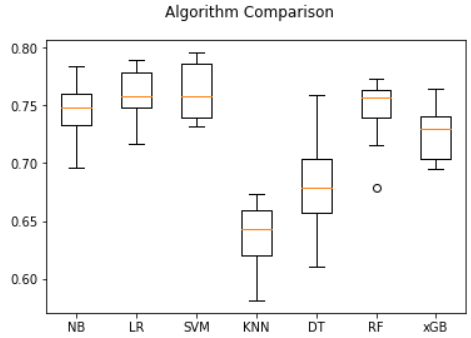

Fig. 6. Accuracy performance comparison of the models

The co-occurrence of psychosocial characteristics, as outlined by the multidimensional framework [68], enables the identification of hate speech in text messages to be robust. This approach addresses the limitations of lexicon-based solutions, which rely primarily on the capacity to detect hate by locating specific domain words in the message, frequently without regard for hate's syntactic patterns, particularly if codeswitching can be utilized to avoid the domain keywords.
The detection of hate speech is conditional on the presence of specific features, which is a common shortcoming of dictionary-based techniques, as the model will not generalize in the absence of these elements.
The primary downside of the general lexicon feature is its sparse vector representation, which results in large numbers of zeros in the vectors. As a result, modeling demands more computer resources, particularly memory, which is a difficulty for typical machine learning techniques.
The conclusion is that the most accurate features and classifiers for recognizing hate speech are PDC features trained with linear SVC classifiers. Additionally, PDC characteristics had a greater effect on accuracy performance when character-level n-grams were used rather than the word- or phrase-level n-grams. This outcome corroborates another study on hate speech [11].

4.4 Model Evaluation and Tuning

This phase focused on determining whether the classification model performed as expected and on determining how to improve the classifier's performance via parameter tuning.
The SVM was found to be the highest performing model out of the nine studied in the studies, as determined by constructing its confusion matrix and determining its F1, precision, and recall score values. The SVM model performed consistently well in terms of precision, recall, and F1 score, all of which were 0.77. The study was particularly impressed with the model's performance on the hypotheses class, namely hate speech, which had a precision score of 0.81, a recall score of 0.85, and an F1 score of 0.83. The complete findings are given in Table 3.

Table 3. Evaluation of SVM model

Class	precision	recall	f1-score	support
0	0.81	0.85	0.83	203
1	0.70	0.71	0.71	226
2	0.79	0.75	0.77	206
accuracy			0.77	635
macro avg	0.77	0.77	0.77	635
Weighted avg	0.77	0.77	0.77	635

A general observation from the normalized confusion matrix in Fig. 7 is that the lower triangle of the matrix had more misclassification than the higher triangle. This suggests that the SVM algorithm was more predisposed than the human coders to label texts as hate speech or offensive.

UNDER PEER REVIEW

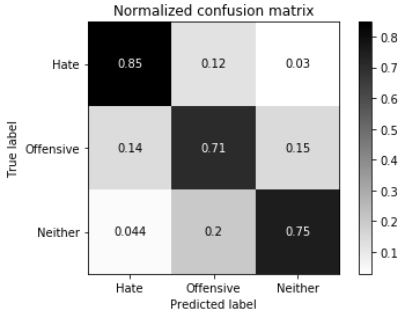

Fig. 7. Confusion matrix for the balanced dataset

According to the first column of the matrix, the model accurately predicted 85 percent (recall) of actual hate speech messages as true positives but misclassified 14% and 4% of offensive and "neither" messages as hate messages, respectively. The misclassification, particularly of hostile communications as false positives, can be explained by the fact that hate speech and offensive messages have some traits. According to the multidimensional framework for hate speech[68], hate speech includes offensive language, although not all offensive statements are inherently hate speech unless they expressly target a protected feature. Additionally, the presence of objectionable lexical phrases would lead the classifier to classify the communication as hate speech, whereas the human annotator would evaluate the context and use hindsight to classify it differently. The misclassification could also be the result of incorrect annotations impacted by the human coder's sensitivity to everyday language use, long-held cultural, religious, and other social belief systems [25, 69].

According to the second column of the matrix, the model correctly predicted 71% of offensive messages, while misclassifying 12% of actual hate speech and 20% of "neither" messages as offensive. This could be explained, once again, by the inherent bias and subjectivity of the human annotator in this work.

The third column of the matrix indicates that the model accurately forecasted 75% of messages as "neither," but misclassified 3% of actual hate speech and 15% of offensive messages as 'neither'.

As a result, the model had the most difficulty in identifying messages classified as offensive, with 12 percent and 20% of real cases of hate speech, respectively, and 'neither' being incorrectly projected as offensive. This, too, might be explained by instructor noise added during annotation as a result of our annotators' differing degrees of sensitivity to what they regarded to be objectionable.

With a soft margin, C=0.1, probability=true, and a Gaussian Radial Basis Function (RBF) kernel, gamma ()=0.1, as the best hyper-parameter values, the SVM classifier outperformed all other classifiers. The C value was chosen as a result of the necessity for a categorization model that would generalize well to various types of hate speech. As a result, the model needed to be trained to exhibit greater tolerance when defining the decision boundary, which is accomplished via machine learning by lowering the penalty associated with model misclassification[69]. The kernel is chosen for SVM assisted in determining the optimum way for the model to build a nonlinear decision boundary based on the features. The gamma () hyper-parameter was critical in setting the decision boundary's sensitivity to new features. A larger gamma value indicates that additional features will exert a greater influence on the decision border, twisting it. As a result, lower values for the soft margin and kernel hyperparameters were optimal for configuring the SVM classifier to handle the otherwise non-linearly separable scenario of text data from social media. Additionally, SVM classifiers are quite resilient and produce amazing predictions when used as models.

4.5 Generalizability of the Classification Model

The subject of model generalizability was central to this research and served as a lens through which all other aims and trials were viewed. Thus, from the start, the study sought a thorough knowledge of the phenomena of hate speech and its conspicuous traits as informed by relevant psychological and sociological theories. This resulted in the development of a multidimensional framework for hate speech, which was utilized to guide the data gathering and annotation operations. Although the data from the 2017 Kenyan presidential elections, which primarily contained ethnic hate speech, was used to train the machine classifier in this work, it is not limited to ethnic hate. To begin, the top-performing classifier in terms of generalizability was trained on a balanced dataset using data from the various studies. By and large, a classifier trained on a balanced dataset with an equal or nearly equal number of class occurrences will not be skewed toward any particular class, in contrast to a classifier trained

UNDER PEER REVIEW

on a dataset that is biased toward the majority class[70]. Second, our model is based on a multidimensional framework for hate speech that is conceptually universal. As a result, it should generalize to different forms of hate speech and in any language, provided that it is retrained with positive examples of that form of hate speech. As a result, our approach was able to positively identify other forms of hate speech in previously unknown messages, such as , " ***Kill*** *all* ***those Muslims*** *to eradicate terrorism*" (Religious hatred); *"**Wtf**! Eastleigh explosion.* ***Wasomali*** *warudi kwao"* (Nationality hatred); *"Thot the 'summerbreak' is over?* ***hawa wazungu*** *waende zao bana! kazi kutuchafulia ma lightskins wetu nkt eyesore galore"* (Racial hatred); *"**Women** are some of the most **corrupt** individuals when placed in positions of power. "* (Gender hatred).
These messages exhibited three major characteristics of hate speech as specified in the conceptual framework for hate speech. These include negative passions such as Kill, Wtf; distancing language through the use of plural pronouns such as that,' *hawa'* (these); and stereotyping by the use of protected characteristics such as Muslims, *Wasomali* (Somalis), *Wazungu* (Whites), and Women. When these characteristics are combined in a single message, the hate speech flag is raised.
Fundamentally, the conceptual framework for hate speech aids in delineating the hypothesis class, h $\epsilon \mathcal{H}$, to which hate speech occurrences can be mapped. As such, the machine learning algorithm's task is to identify the specific hypothesis, h H, that most closely approximates hate speech.
The generalizability of the model also addresses the dynamic nature of language and how to handle future phrases that are not included in the training set. The question here is whether the hypothesis holds for previously unknown cases that were not included in the training set, such as instances in the test or validation data provided via cross-validation. This can be resolved by inducing a class *S* with the property *h* equal to *S*. This means that *S* must contain solely examples of positive hate speech. Alternatively, a broad hypothesis, *G*, might be utilized that encompasses all good examples of hate speech while excluding any incorrect examples. As demonstrated in a recent study, the algorithm may be retrained using a *G*-set that includes examples of the new terms, as well as increasing the margin, which results in an increased distance between the boundary and the nearest occurrences. [71].

5. PDC-BASED CLASSIFICATION MODEL

The study advocates for a novel text categorization framework that employs a mix of psychosocial features (PDCs) based on language connoting negative passion, psychological distancing, and commitment to hate as the major informative concepts for distinguishing subtle forms of hate speech. These qualitative characteristics, which are well established in hate theories such as the duplex theory of hate, provide a rich framework for detecting these elusive hostile utterances, particularly when concealed by codeswitching, which previous methods were unable to detect.
The classification model based on PDCs is supervised machine learning-based. It is divided into three major components: data pre-processing, feature engineering, and model construction and evaluation. These are depicted in Figure 8. The component of data preparation is divided into two subcomponents, namely data annotation, and data preprocessing. As is customary in supervised machine learning, the input to the PDC-based model is labeled data that can be used to solve binary classification or multi-class classification problems. This means that the data can be annotated with a maximum of two labels, such as positive or negative in the case of binary classification, or with more than two labels, such as high, medium, and low in the case of multi-class classification. Human annotators frequently label the raw data input according to some annotation strategy. The annotation scheme used in this investigation is based on a strong theoretical foundation, as detailed in our prior study [1] and as illustrated in the first zoomed-out component in Fig. 8
For instance, the multi-dimensional framework based on the PDC can capture the use of devaluation in a codeswitched message such as, "*Do not make music with* ***those cockroaches****, hiyo ndiyo dawa ['that's the medicine needed'] to silence* ***them***". In Kenya, certain ethnic devaluation terms are well recognized and frequently used by the in-group membership to refer to out-groups. For instance, the term, "foreskins", or "fish", is frequently used to infer and denigrate the Luo ethnic group, which does not practice circumcision traditionally. The usage of stereotypes translates into the subtle use of harsh words without the use of overtly nasty lexicons. For instance, the Kikuyu, Kamba, and Kisii ethnic groups are referred to by compound phrases such as, "*money lovers*", "*tire thieves*", or "*night runners*". These nuanced kinds of hate speech, particularly when codeswitching is used, frequently pass unnoticed by conventional filters.
The data pre-processing sub-unit is responsible for tokenizing and cleaning annotated text, which is frequently noisy. Standard data cleaning procedures are followed, including the elimination of punctuation, duplication, empty strings, non-alphanumeric characters, lowercasing, stemming, and stopword removal. Unlike traditional models, which remove all pronouns uniformly throughout the

UNDER PEER REVIEW

preprocessing step, the PDC-based model keeps pronouns when deleting Stopwords. This is because the presence of pronoun dichotomies in a message has previously been demonstrated to be informative in signaling "othering" language[41][58], which is a hate speech notion associated with psychological distancing. For instance, " ***We*** *shall not allow* ***them*** *to cross river Tana. Punda hao!"*

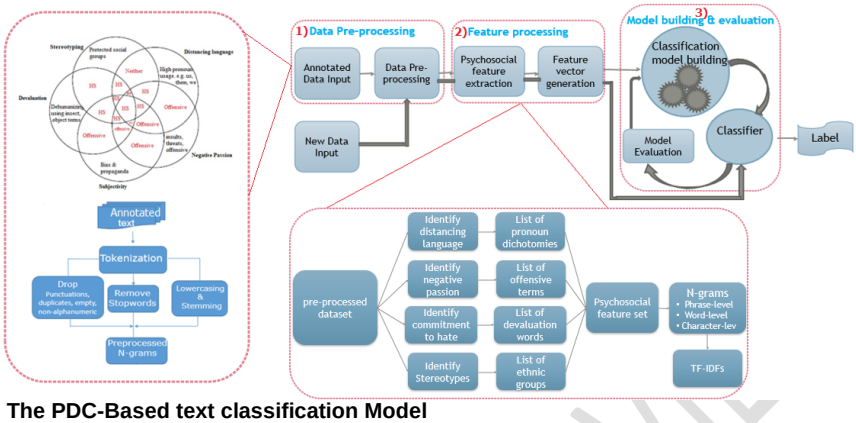

Fig. 8: The PDC-Based text classification Model

The first component produces a pre-processed dataset that has been de-noised and normalized using lowercasing and stemming. As a result, the dimensionality of the input is significantly reduced in contrast to the initial raw annotated input. However, textual data offers a barrier because the major attributes are frequently the words or tokens. As a result, this feature set converts into a high-dimensional feature space, with a sparse input vector to the machine learning method in component 3, needing additional processing effort and memory. This difficulty is addressed by the PDC-based model in component 2, which consists of the PDC vocabulary learning and feature vector generation subcomponents. The first subcomponent filters the pre-processed dataset by extracting vocabulary indicative of psychosocial distancing, negative passion, hatred commitment, and stereotyping to create their respective lists. The seed attributes for each list were primarily inspired by qualities that are effective in similar challenges in the literature [4, 15, 51] and by psychological word categories in the Linguistic Inquiry Word Count analyzer [67]. Besides, the respective feature categories were populated with terms that were connected to the classes in the Latent Dirichlet Allocation topic models that were constructed automatically. The five feature categories were structured structurally in a table based on word families, with the first column indicating the word family, successive columns holding the word forms or features, and rows containing the meanings. Bootstrapping enables the addition of new words or terms with similar meanings in other languages, i.e. codeswitched words, to the corresponding feature columns. As described in Table 4, the structural shape is as follows.

Table 4. PDC Conceptual lookup table

Word-Family	**Word Form (Features)**					
	Feature1			**Feature …**	**…**	**Feature n**
Negative **P**assion	F_{L1}	F_{L2}	F_{Ln}			
Distancing						
Commitment (Devaluation)						
(Stereotyping)						
(Subjectivity)						

This psychosocial feature set, PDC, can then be analyzed at multiple levels, such as phrase, word, or character, and translated into numerical feature vectors, in this case, TF-IDFs. As a feature selection and representation method, TF-IDF is designed to rank tokens according to their importance across the corpus, penalizing tokens at either extreme, i.e., words that are extremely frequent or extremely infrequent across documents, for being deemed unimportant or outliers. As a result, the resulting TF-IDF feature vector derived from the high-level PDC feature set is dense, making it a more acceptable input for component 3's classification model development. A suite of machine learning algorithms is trained on the TF-IDF input vector to construct their respective classifiers, based on their performance in recognizing hate speech in prior similar studies. Often, it is difficult to predict which machine learning algorithm will be optimal for a classification task in advance. As a result, it is usual practice in

machine learning to experiment with multiple algorithms, beginning with the simplest, to choose the optimal one for the particular machine learning problem[70]. The best classifier model is reviewed and tested based on its performance and results. There are two methods for evaluating. To begin, the correlation between the features and the class, i.e., the text vector and the label column value, is calculated using the Chi-Square feature scoring method. Second, the confusion matrix is utilized to calculate the precision, recall, and, ultimately, the accuracy of the trained model when applied to the testing dataset. Finally, the pre-processing sub-component is given a fresh text message as input. It is not required to include annotations. It must, however, pass through the feature processing component and be converted to its TF-IDF vector representation. Following that, the vector is passed straight to the classifier, which outputs the projected class label. This is illustrated in Figure 8.

In summary, tests were done to confirm our method of utilizing psychosocial concepts drawn from established hate theories in psychology and sociology to construct a novel psychosocial feature set, dubbed PDC. Following that, the PDC feature set was converted to TF-IDF vectors to train a classification model for detecting nuanced kinds of hate speech, particularly in codeswitched data. Our classifier performed significantly better than the baseline, which was the human inter-rater reliability score for the same annotated dataset, by over 27% in classification accuracy. The classifier's generalization was subsequently evaluated using an unseen dataset of racist, religious, and nationality-based abusive comments. The results were comparable to those obtained using state-of-the-art baseline classifiers for the classification of similar types of hate speech. However, given the variety of datasets used in this study, particularly the emphasis on codeswitched data, it would be unrealistic to directly compare to publically available monolingual datasets. Additionally, the study indicated the psychosocial characteristics' capacity to generalize to other forms of hate speech, such as racist insults.

6. **CONCLUSION**

This study makes three contributions. First, it provides a gold-standard annotated dataset that may be used for comparative studies by other researchers. Second, the study provided an empirical framework and methodology for recognizing nuanced kinds of hate speech in brief text messages that are founded in theory. Thirdly, this approach was important in the development of a text classification model capable of effectively generalizing to various forms of hate speech on social media. Subsequently, the classifier's outputs could be utilized to influence evidence-based judgments by relevant security authorities and data-driven policy formation addressing the monitoring of hate speech on social media during future presidential elections in Kenya.

The psychosocial feature set analyzes language use concerning the concepts of psychosocial distancing, negative passion, commitment to hatred, stereotyping, and hate as a story to identify nuanced forms of hate speech that conventional methods were unable to detect, particularly in codeswitched data. These principles are well-supported by the duplex theory of hate[54] and are summarized in fig.4. Additionally, the study presents a straightforward and effective method for qualitatively identifying and analyzing hate speech in short text documents through the use of human-readable high-level psychosocial features, specifically PDC-based features, which can be mapped to machine-readable lower-level features such as Term Frequency-Inverse Document Frequency (TF-IDF) and one-hot encoding vectors for training a machine classifier. Previous research on hate speech identification has relied on lexical and other natural language processing (NLP) elements. However, these characteristics alone are insufficient to capture the full range of hate speech in codeswitched messages. As a result, classifier models that explicitly use these traditional features will underperform, producing a high number of false negatives, in contrast to how hate is expressed in social media writings.

The psychosocial (PDC) elements are intended to be useful in two critical aspects. To begin, the feature set should be sufficiently informative to improve classification performance. Second, the PDC feature set is significantly less than that of standard approaches that use the TF-IDF to represent the complete input lexicon. This drastically lowers the sparseness and dimensionality of the original features, making PDC a great feature selection technique with a dense input vector length, in contrast to the general lexicon's sparse input vector. Additionally, the efficiency of the PDC design as a qualitative feature selection strategy for codeswitched text categorization of nuanced kinds of hate speech will benefit general machine classification efforts.

The comparison of the various features used to train the nine machine learning models in this study reveals that the PDC features, which employ character-level n-grams, are the most discriminative in classifying hate speech in codeswitched text messages. The best performance was obtained with n=3 to 5 characters, respectively when the SVM classifier was used. This could be explained by the great degree of language independence of character n-gram features, which enables feature extractors to

UNDER PEER REVIEW

be easily portable between languages[72]. Additionally, this characteristic has been demonstrated to be the most significant in the authorship categorization task [73]. The disadvantage of character n-grams is that they increase the dimensionality of the feature space, which is particularly noticeable when working with very big datasets. Nonetheless, their performance is superior to that of standard n-grams when used with conventional machine learning methods and moderately specified computer hardware.

Future work will consider moving away from the current discrete representation of PDC features in which words exist as atomic symbols and toward a distributed representation in which dense vectors could be used to represent word families to accommodate synonyms, hypernyms, and codeswitching in their context words.

REFERENCES

[1] A. Des Forges, "Leave None To Tell The Story: Genocide in Rwanda," *New York Hum. Rights Watch*, 1999.

[2] S. Benesch, "Dangerous Speech: A Proposal to Prevent Group Violence," 2012.

[3] L. Silva, M. Mondal, D. Correa, F. Benevenuto, and I. Weber, "Analyzing the Targets of Hate in Online Social Media," in *Tenth International AAAI Conference on Web and Social Media*, 2016, pp. 687–690.

[4] R. Hatzipanagos, "How online hate turns into real-life violence," *The Washington Post*, Washington, 30-Nov-2018.

[5] R. Ajulu, "Politicised Ethnicity, Competitive Politics and Conflict in Kenya: A Historical Perspective," *Afr. Stud.*, vol. 61, no. 2, pp. 251–268, 2002.

[6] P. Makori, "Whatsapp admins face jail in crackdown to curb hate-speech," *Business Today*, 17-Jul-2017.

[7] S. Madonsela, "A critical analysis of the use of code-switching in Nhlapho's novel Imbali YemaNgcamane," *South African J. African Lang.*, vol. 34, no. 2, pp. 167–174, 2014.

[8] E. Ombui and L. Muchemi, "Wiring Kenyan Languages for the Global Virtual Age: An audit of the Human Language Technology Resources," *Int. J. Sci. Res. Innov. Technol.*, vol. 2, no. 2, pp. 35–42, 2015.

[9] M. Karani, E. Ombui, and A. Gichamba, "The Design and Development of a Custom Text Annotator," in *IEEE Africon*, 2019.

[10] A. I. Ansaldo, K. Marcotte, L. Scherer, and G. Raboyeau, "Language therapy and bilingual aphasia: Clinical Implications of psycholinguistic and neuroimaging research," *J. Neurolinguistics*, vol. 21, 539–55, 2018.

[11] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? predictive features for hate speech detection on Twitter.," in *In Proceedings of NAACL-HLT*, 2016, pp. 88–93.

[12] G. Priya, K. Aditi, T. Richa, A. Mayank, B. Sohail, and J. Vishal, "A Proposed Framework to Analyze Abusive Tweets on the Social Networks," *Int. J. Mod. Educ. Comput. Sci.*, vol. 10, no. 1, pp. 46–56, 2018.

[13] W. Warner and J. Hirschberg, "Detecting hate speech on the world wide web," in *Language in Social Media (LSM 2012)*, 2012.

[14] I. Kwok and Y. Wang, "Locate the hate: Detecting tweets against blacks," *AAAI*, 2013.

[15] D. N. Gitari, Z. Zuping, H. Damien, and J. Long, "A lexicon-based approach for hate speech detection.," *J. Multimed. Ubiquitous Eng.*, vol. 4, no. 10, pp. 215–230, 2015.

[16] E. Spertus, "Smokey: Automatic recognition of hostile Messages," in *IAAI*, 1997.

[17] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "Detecting offensive language in social media to protect adolescent online safety," in *The fourth ASE/IEEE international conference on social computing (SocialCom 2012)*, 2012.

[18] D. K, B. Jones, C. Havasi, H. Lieberman, and R. Picard, "Common sense reasoning for detection, prevention, and mitigation of cyberbullying.," *ACM Trans Interact Intell Syst*, vol. 3, no. 2, 2012.

[19] C. Van Hee and G. De Pauw, "Automatic Detection and Prevention of Cyberbullying," in *The First International Conference on Human and Social Analytics*, 2015.

[20] S. Agarwal and A. Sureka, "Using KNN and SVM Based One-Class Classifier for Detecting Online Radicalization on Twitter," in *The 11th International Conference on Distributed Computing and Internet Technology*, 2015, pp. 431–442.

[21] M. Last, A. Markov, and A. Kandel, "Multi-lingual Detection of Terrorist Content on the Web," in *International Workshop on Intelligence and Security Informatics*, 2006.

UNDER PEER REVIEW

[22] E. Lozano, J. Cedeno, G. Castillo, F. Layedra, H. Lasso, and C. Vaca, “Requiem for online harassers: Identifying racism from political tweets,” in *Fourth International Conference on eDemocracy & eGovernment (ICEDEG)*, 2017.
[23] S. Tulkens, L. Hilte, E. Lodewyckx, B. Verhoeven, and W. Daelemans, “The Automated Detection of Racist Discourse in Dutch Social Media,” *CoRR, abs/1608.08738*, 2016.
[24] T. Davidson, D. Warmsley, M. Macy, and I. Weber, “AutomatedHateSpeechDetectionandtheProblemofOffensiveLanguage,” in *ICWSM*, 2017.
[25] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, “Deep Learning for Hate Speech Detection in Tweets,” in *2017 International World Wide Web Conference Committee*, 2017.
[26] P. Fortuna, “Automatic detection of hate speech in text: an overview of the topic and dataset annotation with hierarchical classes,” University of Porto, 2017.
[27] P. Burnap and M. L. Williams, “Cyber Hate Speech on Twitter: An Application of Machine Classification and Statistical Modeling for Policy and Decision Making,” *Policy & Internet*, vol. 2, no. 7, pp. 223–242, 2015.
[28] C. Nobata, J. Tetreault, A. Thomas, Y. Mehrdad, and Y. Chang, “Abusive Language Detection in Online User Content,” in *25th International Conference on World Wide Web*, 2016, pp. 145–153.
[29] P. . O’Sullivan and A. . Flanagin, “Reconceptualizing ‘flaming’ and other problematic messages,” *New Media Soc.*, vol. 5, pp. 69–94, 2003.
[30] A. Mahmud, K. . Ahmed, and M. Khan, “Detecting Flames and Insults in Text,” in *In Proceedings of the 6th International Conference on Natural Language Processing*, 2008.
[31] A. Razavi, D. Inkpen, S. Uritsky, and S. Matwin, “Offensive Language Detection Using Multi-level Classification,” *Springer*, p. 1627, 2010.
[32] I. Chaudhry, “Hashtagging hate: Using Twitter to track racism online,” *First Monday 20(2)*, 2015. .
[33] S. Liu and T. Forss, “New classification models for detecting Hate and Violence web content,” in *7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*, 2015, pp. 487–495.
[34] A. Gaydhani, V. Doma, S. Kendre, and L. Bhagwat, “Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach,” 2018.
[35] M. Hasanuzzaman, G. Dias, and A. Way, “DemographicWordEmbeddingsforRacismDetectiononTwitter,” in *Proceedings of The 8th International Joint Conference on Natural Language Processing,* 2017, pp. 926–936.
[36] N. Djuric, J. Zhou, M. Morris, Robin Grbovic, V. Radosavljevic, and N. Bhamidipati, “Hate speech detection with comment embeddings,” in *In Proceedings of the 24th InternationalConferenceonWorldWideWeb(WWW2015),* 2015, pp. 29–30.
[37] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” 2014. [Online]. Available: https://nlp.stanford.edu/pubs/glove.pdf. [Accessed: 19-Sep-2019].
[38] Z. Mossie and J.-H. Wang, “SOCIAL NETWORK HATE SPEECH DETECTION FOR AMHARIC LANGUAGE,” in *COMIT*, 2018, pp. 41–55.
[39] A. Al-Hassan and H. Al-Dosari, “Detection of Hate Speech in Social Networks: A Survey on Multilingual Corpus,” in *6th International Conference on Computer Science and Information Technology*, 2019.
[40] D. Gamal, M. Alfonse, M. E.-H. El-Sayed, and A.-B. M.Salem, “Twitter Benchmark Dataset for Arabic Sentiment Analysis,” *Int. J. Mod. Educ. Comput. Sci.*, vol. 11, no. 1, pp. 33–38, 2019.
[41] W. Alorainy, P. Burnap, H. Liu, and M. L. Williams, “‘The Enemy Among Us’: Detecting Cyber HateSpeech with Threats-based Othering Language Embeddings,” *ACM*, 2019.
[42] K. N. B. of Statistics, “2019 Kenya Population and Housing Census Volume I: Population by County and Sub-County,” 2019.
[43] H. Kim, S. Jang, Mo, S.-H. Kim, and A. Wan, “Evaluating Sampling Methods for Content Analysis of Twitter Data,” *Sage*, 2018.
[44] A. E. Kim, H. M. Hansen, J. Murphy, A. K. Richards, J. Duke, and J. A. Allen, “Methodological Considerations in analyzing Twitter data,” *J. Natl. Cancer Inst.*, vol. 47, pp. 140–146, 2013.
[45] P. . Cavazos-Rehg *et al.*, “A content analysis of depression-related tweets,” *Comput. Hum. Behav.*, vol. 54, pp. 351–357, 2016.
[46] C. Shearer, *The CRISP-DM model: the new blueprint for data mining*. 2000.
[47] F. Provost and T. Fawcett, *Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking*, First Edit. O’Reilly Media, Inc., 2013.
[48] Z. Waseem, “Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter,” in *EMNLP Workshop on NLP and CSS*, 2016, pp. 138–142.

UNDER PEER REVIEW

[49] W. Warner and J. Hirschberg, “Detecting Hate Speech on the World Wide Web,” in *Language in Social Media (LSM 2012)*, 2012.
[50] P. Burnap and M. L. Williams, “Us and them: identifying cyber hate on Twitter across multiple protected characteristics.,” *EPJ Data Sci.*, 2016.
[51] R. . King and G. M. Sutton, “High Times for Hate Crime: Explaining the Temporal Clustering of Hate Motivated Offending,” *Criminology*, vol. 51, no. 4, pp. 71–94, 2013.
[52] J. Brownlee, *Deep Learning for Natural Language Processing*, V1.2. 2018.
[53] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet allocation,” *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.
[54] R. Sternberg and K. Sternberg, “The Duplex Theory of Hate I: The Triangular Theory of the Structure of Hate. In The Nature of Hate,” *Cambridge Univ. Press*, pp. 51–77, 2008.
[55] M. Elsherief, V. Kulkarni, D. Nguyen, W. Wang, and E. Belding, “Hate lingo: A target-based linguistic analysis of hate speech in social media,” in *12th International AAAI Conference on Web and Social Media*, 2018, pp. 42–51.
[56] N. Coupland, “‘Other’ representation, Society and Language.” John Benjamins Publishing, 2010.
[57] G. R. Semin, “Linguistic Markers of Social Distance and Proximity.” 2009.
[58] M. Cikara, M. M. Botvinick, and S. T. Fiske, “Us versus them: Social identity shapes neural responses to intergroup competition and harm,” *Psychol. Sci.*, vol. 22, no. 3, pp. 306–313, 2011.
[59] N. Haslam, “Dehumanization: An integrative review,” *Personal. Soc. Psychol. Rev.*, vol. 10, pp. 252–64, 2006.
[60] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
[61] K. Krippendorff, “Computing Krippendorff’s Alpha-Reliability,” *University of Pennsylvania ScholarlyCommons*, 2011. [Online]. Available: mhttp://repository.upenn.edu/asc_papers/43.
[62] W. Clyne, S. Pezaro, K. Deeny, and R. Kneasfsey, “Using Social Media to Generate and Collect Primary Data: The #ShowsWorkplaceCompassion Twitter Research Campaign,” *JMIR Public Heal. Surveill*, vol. 4, no. 2, p. e41, 2018.
[63] V. Dijk and A. Teun, “Discourse and racism, The Blackwell companion to racial and ethnic studies,” pp. 145–159, 2002.
[64] S. Sood, J. Antin, and E. Churchill, “Profanity use in online communities,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1481–1490.
[65] P. Fortuna, L. da Silva, Jo˜ao Rocha Soler-Company, Juan Wanner, and S. Nunes, “A Hierarchically-Labeled Portuguese HateSpeech Dataset,” in *Proceedings of the Third Workshop on Abusive Language Online*, 2019, pp. 94–104.
[66] B. Ross, M. Rist, G. Carbonell, B. Cabrera, N. Kurowsky, and M. Wojatzki, “Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis,” *arxiv:1701.08118*, vol. 1, 2017.
[67] Y. R. Tausczik and J. W. Pennebaker, “The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods,” *J. Lang. Soc. Psychol.*, vol. 1, no. 29, 2010.
[68] E. Ombui, L. Muchemi, and P. Wagacha, “Building and Annotating a Codeswitched Hate Speech Corpora,” *Int. J. Inf. Technol. Comput. Sci.*, no. 3, pp. 33–52, 2021.
[69] L. Chen, “Support Vector Machine — Simply Explained,” *Towards Data Science*, 2019. [Online]. Available: https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496. [Accessed: 02-Apr-2020].
[70] J. Brownlee, *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch*. 2016.
[71] E. Alpaydin, *Introduction to Machine Learning*, 2nd Editio. London: The MIT Press, 2010.
[72] F. Peng, D. Schuurmans, V. Keselj, and S. Wang, “Language independent authorship attribution with character-level n-grams,” in *10th Conference of the European Chapter of the Association for Computational Linguistics*, 2003, pp. 267–274.
[73] J. Kruczek, P. Kruczek, and M. Kuta, “Are n-gram Categories Helpful in Text Classification?,” in *International Conference on Computational Science*, 2020, pp. 524–537.

UNDER PEER REVIEW