ISSN: 2710-2599

International Journal of Data Science and Big Data Analytics

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

An Optimal WordNet Based Emotional Word Extraction and Hybrid Deep Learning Classifier for Sentiment Analysis

Stephen Obare[1*], Abejide Ade-Ibijola[2] and Kennedy Ogada[3]

[1]School of Computing & Information Technology Jomo Kenyatta University of Agriculture and Technology Nairobi, Kenya. E-mail: smobareo@gmail.com

[2]Formal Structures, Algorithms and Industrial Applications Research Cluster School of Consumer Intelligence and Information Systems, University of Johannesburg, South Africa. E-mail: abejideai@uj.ac.za

[3]School of Computing & Information Technology Jomo Kenyatta University of Agriculture and Technology Nairobi, Kenya. E-mail: kodhiambo@scit.jkuat.ac.ke

Article Info

Volume 3, Issue 1, May 2023

Received : 24 August 2022

Accepted : 17 March 2023

Published : 05 May 2023

doi: 10.51483/IJDSBDA.3.1.2023.25-44

Abstract

Everyday, over one billion social media text messages are generated worldwide, which is a rich source of data that can lead to improvements in the lives of citizens through evidence-based decision making. Twitter is rich in such data but there are number of challenges in processing tweets with respect to volume, speed, ambiguity of the language in which tweets are written, which is an information extraction problem, in the domain of Natural Language Processing (NLP). While there is a growing interest in sentiment analysis for detecting emotions from tweets, there are no major efforts for detecting emotions that are disguised in tweets based on context of word usage, which is important for tasks such as identification of events such as hate speech, mental health related disorders. This paper presents a novel approach to context-based hate speech detection based on an optimal WordNet. Taking a modified Mayfly Optimization algorithm (MMO), we pre-process tweets and normalize the data using an NLP pipeline. We argument an improved Horse Herd Optimization algorithm (HHO) with WordNet and SentiWordNet to compute a tweet sentiment polarity. A hybrid Deep Belief Artificial Neural Network (hybrid DB-ANN) is then used to classify tweets. The performance of the proposed approach is compared to best known sentiment analysis algorithms using three standard benchmark datasets: Crowdflower-1, Crowdflower-2, and Kaggle Twitter. We demonstrate that the proposed approach outperforms industry benchmarks as it achieves over 90% in terms of accuracy, precision, recall, and F-measure.

Keywords: *Natural Language Processing, Sentiment Analysis, WordNet, SentiWordNet, Tweet classification, Emotional words*

* *Corresponding author: Stephen Obare, School of Computing & Information Technology Jomo Kenyatta University of Agriculture and Technology Nairobi, Kenya. E-mail: smobareo@gmail.com*

1. Introduction

The rapid growth of the internet has made social networking platforms vital for conveying emotions widely (Khan *et al.*, 2016). Micro blogging sites have created vast amounts of unstructured data every second as citizens express themselves through text, images, music, and video (Hung and Chen, 2016; Khan *et al.*, 2016). To understand human psychology through emotional word detection, data must be processed as quickly as it is generated through sentiment analysis (Khan *et al.*, 2016). While sentiment analysis identifies a tweet as either positive, negative, or neutral (Hung, 2017; Gutiérrez *et al.*, 2017), emotions like joy, sorrow, fear, rage, and surprise that originate from personal experiences and interactions with the environment generate a lot of sentiment-rich data (AL-Sharuee *et al.*, 2018). Psychologically, expressed emotions is crucial in determining the feelings of a person either as an individual or towards a product or a service (Jha *et al.*, 2018; Akilandeswari and Jothi, 2018) hence the need to mine such information for personal safety or business purposes. The ambiguity of natural languages and the fact that feelings can be disguised in tweets pose a challenge when processing such data (Medagoda *et al.*, 2015).

Significant research has gone into automating detection of online occurrences such as hate speech where most researchers employ NLP, deep learning, and machine learning strategies (Sharma *et al.*, 2022; Cruz *et al.*, 2022). Hate speech detection from different languages has received significant attention within the research community. An ensemble learning technique which uses several feature spaces to learn from unintentional biased assessment measures was proposed in Nascimento *et al.* (2022); Plaza-del-Arco *et al.* (2021) for hate speech identification. A three-class instance-based approach in Pronoza *et al* (2021) was designed to detect ethnic hate speech on Russian social media text with a new three-class strategy. Multilingual language representations was used in joint-learning models (Pamungkas *et al.*, 2021). A Naïve Bayes classifier fused with term frequency inverse document frequency system was proposed in Ayo *et al.* (2021) to automatically cluster real-time tweets into topic clusters. Deep learning-based data augmentation which combines back translation and paraphrasing for analysis of word-embedding-based hate speech classifications was proposed in Beddiar *et al.* (2021). The model used back translation based on an encoder–decoder system that was trained on a huge dataset for machine translation. Kocon *et al.* (2021) proposed a group-based and individual demographics multimodal model for hate speech detection. An encoder decoder-based machine learning model which classifies Bengali Facebook comments was proposed in Das *et al.*(2021). In all the above cases, the models do not achieve detection accuracy, precision, recall, and F-measure of above 80% hence the need to improve on the existing work.

One technique that can be used to improve the detection accuracy, precision, recall, and F-measure from unstructured tweets is augmenting best in class algorithms with a more intelligent preprocessing engine and WordNet. In this paper, we have designed a novel technique for sentiment analysis using WordNet for contextual analysis to aid in emotional word extraction and a hybrid deep learning classifier for improved detection accuracy, precision, recall, and F-measure. We have:

1. Modified MO algorithm for data pre-processing;
2. Used an NLP pipeline to normalize tweets;
3. Extracted emotional words from tweets using WordNet and SentiWordNet dictionary;
4. Computed sentiment polarity using an improved HHO algorithm; and
5. Improved the detection accuracy of emotional tweets using hybrid DB-ANN.

The rest of this paper is organized as follows. Section 2 presents the problem statement and background to this study, and the review of relevant literature. Section 3 presents the methodology and system design of the proposed approach. In Section 4, we present the implementation and simulate the proposed technique's working function. Section 5 provides a comparison of the proposed technique and existing sentiment analysis techniques while the last Section presents the conclusions and future work.

2. Problem Statement and Background

This section introduces the problem solved in this paper and highlights the motivations behind this work. First, we start with the problem addressed in this paper and the motivation behind this work.

2.1. Problem Statement and Motivation

Existing studies have either used a combination of lexical semantic dictionaries with ML or hybrid ML hybrid deep learning algorithms for Twitter sentiment analysis as shown in Table 1. Techniques based on SentiWordNet, WordNet and ML suffer from several challenges including design complexities due to data dimentionality because of data sparsity (Miranda *et al.*, 2019; Asgarian *et al.*, 2018; Khan *et al.*, 2017; Dehkharghani *et al.*, 2016). Techniques based on MLs and hybrid MLs (Dessì *et al.*, 2021; Qureshi and Sabih, 2021; Baydogan and Alatas, 2021; Plaza-Del-Arco *et al.*, 2021; Kapil and Ekbal, 2020; Mossie and Wang, 2020), among other shortcomings include their inability to capture underlying semantics of words.

Pattern-based techniques (Watanabe *et al.*, 2018) which use hateful terms and phrases combined with patterns and sentiment-based features, to identify hate speech automatically produced an average of 78.4% accuracy for hateful, offensive, and clean, and 87.4% accuracy for classifying tweets as clean or non-offensive or offensive. This impressive performance is however only achieved with English language dataset while hate speech detection requires not only the evaluation words but also the context of usage. Existing algorithms assign numeric scores to words and fail to look at the context of the word in the whole sentence. We propose in this paper a technique that assigns meaning to words based on context and modify existing algorithms to achieve better accuracy in detection.

2.2. Background

For sentiment analysis, the development of Twitter data availability and Machine Learning (ML) models in past led several academics to use a ML model. SentiWordNet and ML were used to assess Indonesian tweets for election mood. The Naïve Bayes classification algorithm achieved 71.37% for Prabowo and 74.94% accuracy for Joko Widodo.

FerdowsNet (Asgarian *et al.*, 2018) analyzed Persian emotion polarity. The findings of mapping to SentiWordNet and semi-supervised learning were compared. Extracted features were used to classify

Table 1: Summary of Research Gaps

Ref.	Methodology	Datasets	Findings/ Measurements	Remarks
Miranda (2019)	SentiWordNet and machine learning	Twitter content	Recall, precision, accuracy	Richer, less dependable, and noisier
Asgarian *et al.* (2018)	WordNet and semi-supervised learning	Persian text reviews	F-score	Design complexity due to data dimensionality
Khan *et al.* (2017)	WordNet and semi-supervised learning	Kaggle twitter	Accuracy and F-score	Unable to capture the underlying semantics
Dehkharghani *et al.* (2016)	Turkish WordNet	Turkish text	F-score and accuracy	Not able to achieve high detection rate
Dessì and Diego (2021)	WordNet and supervised learning	Kaggle twitter	Precision, Recall, F-score, AUC	High sparsity and dimensional features
Qureshi and Sabih (2021)	CAT boost for hate speech detection	Hatebase twitter	F-score and Accuracy	Unable to capture the underlying semantics
Baydogan and Alatas (2021)	ALO and MFO algorithm for hate speech detection	Kaggle twitter	Accuracy	Redundant words confuse loss of accuracy
Plaza-Del-Arco *et al.* (2021)	NLP for hate speech detection	Spanish tweets	F-score and Accuracy	Highly skewed data cause loss in accuracy
Kapil and Ekbal (2020)	SP-MTL for hate speech detection	Kaggle twitter database	F-score and Accuracy	Affected by class skew problems
Mossie and Wang (2020)	GRU and RNN for hate speech detection	Facebook, crawled	F-score and Accuracy	Less attention to detecting white supremacist content

sentiment. Using well-known feature selection methodologies and cutting-edge machine learning algorithms, Persian text reviews were sentiment-classified.

A combination of ML with lexicon-based system was developed using a semi-supervised sentiment analysis strategy (Khan *et al.*, 2017). Sentiment scores were modified using information gain and cosine similarity. The first Turkish polarity database which is comprehensive, assigned objectivity, negativity, positivity scores to every Turkish WordNet synset is SentiTurkNet (Dehkharghani *etal.*, 2016). The presented technique yielded better accurate polarity scores than direct SentiWordNet translation, which was suggested by Turkish assessment results. Polarity resources that indicated word emotion were widely utilized.

A system was developed to create lexicons irrespective of the target domain and input classes (Dessì and Diego, 2021). They employed existing natural language processing, WordNet disambiguation tools, and methodologies. The framework accepted an annotated text collection with a defined number of classes. It generated WordNet word senses with weights. Using produced lexicons in an emotion detection challenge confirmed the framework's efficacy.

Exploring text mining features to forecast the various types of hatred for each class, two classes of features were analyzed for problem similarity. Baseline features were the most effective of similar research. Latent semantic analysis (LSA) (Qureshi and Sabih, 2021) lowers CAT Boost, non-linear and complicated models and executed well. Nonlinear models were utilized for classification, and CAT Boost was the best across all datasets.

A metaheuristic-based automated hate speech detection system (Baydogan and Alatas, 2021) was presented to improve results. Moth Flame Optimization (MFO) and Ant Lion Optimization (ALO) were developed for hate speech identification. It required an effectual representation technique and adaptable fitness function. Many indicators could be incorporated into a fitness function to improve them individually. Using Bag of Word (BoW) and document vector (Word2Vec), they extracted features.

To identify hate speech in Spanish tweets, multi-task technique used shared emotional knowledge (Plaza-Del-Arco *et al.*, 2021). Combining emotional knowledge and polarity improved hate speech detection across datasets. Experiments on two benchmark corpora revealed the technique outperformed Single Task Learning (STL). A detailed knowledge transfer study demonstrated that emotion classification and polarity tasks enabled transformer-based model recognized hate speech effectively utilizing emotional information.

Deep multi-task learning (MTL) combined knowledge from numerous related classification tasks to enhance individual task performance (Kapil and Ekbal, 2020). Multi-task model used shared-private method. Five datasets revealed that MTL framework attained improved macro-F1 and weighted-F1 performance.

A hate speech detection technique was developed for detecting hate speech against vulnerable minorities on social media. Spark posts were automatically collected and preprocessed, then n-grams and Word2Vec performed feature extraction. Hate speech identification and classification was done using deep learning methods as Recurrent Neural Networks (RNNs) and Gated Recurrent Unit (GRU) (Mossie and Wang, 2020). Clustering hate words using Word2Vec predicted the target ethnic group.

2.3. Summary of Problem Statement and Background

In this section we have presented the problem and justified why it is an important problem to solve, and why we think our approach is appropriate for the solution. We have also presented the review of related work and the gap in existing solutions. In the next section, we present the design of the proposed solution.

3. System Design

In this section, we present the design of our proposed model use for detecting emotions, based on tweet data extracted from Twitter.

First, we introduce a modified MO algorithm for pre-processing raw tweets, normalize the pre-processed tweets using NLP pipeline, extract emotion words using WordNet and SentiWordNet, assign a numeric score to each word within a tweet. The next step involved modifying HHO algorithm to compute sentiment polarity value for each tweet and finally classify the tweets using a hybrid DB-ANN classifier as shown in Figure 1.

We discuss each step in details in and show how the algorithms have been improved to achieve an optimized solution to the problem of detecting emotional words in the subsequent subsection.

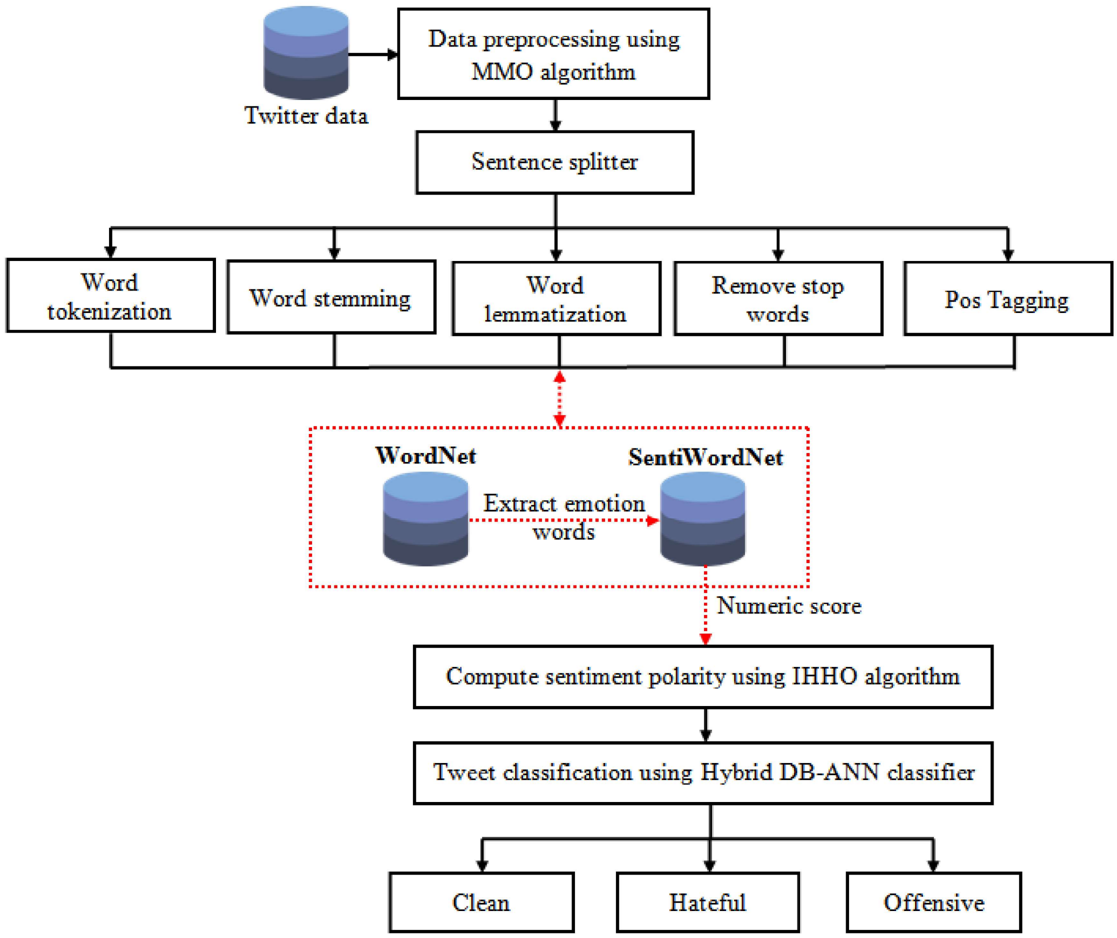

Figure 1: Proposed System Design

3.1. A Modified Mayfly Optimization Algorithm for Tweet Pre-processing

Pre-processing is a vital step when dealing with data which is both noisy and colloquial in nature such as Twitter. We introduced a modified Mayfly Optimization (MO) algorithm, which is inspired by Mayfly Algorithm (MA) that incorporates the benefits of both swarm intelligence and evolutionary algorithms to isolate and delete repeated tweets, hashtags and unwanted artifacts. MA shows improved exploration and convergence capabilities and has a more noteworthy likelihood of deciding the global optimal solution. The Mayflies in swarms for MA are isolated into female as well as male. The male Mayflies are generally stronger thus better for optimizing process. In MMO algorithm, situations are refreshed as per ongoing positions pi(n) and speed vi(n) at the ongoing cycle:

$$p_i^{(n+1)} = p_i(n) + v_i^{(n+1)} \qquad ...(1)$$

The speed is refreshed by ongoing fitness values $f(x_i)$ and the recorded best fitness values in directions $f(x_{hi})$. If $f(x_{hi}) < f(x_i)$, formerly, the male mayflies would refresh their speeds as per its ongoing speeds, along with the distance among them and the worldwide best position, the recorded best directions as follows:

$$V_i(n+1) = h.v_i(n) + b_1\, e_p^{-\beta r^2}\left[xh_i - x_j(n)\right] + b_2\, e_p^{-\beta r^2}\left[x_g - x_i(n)\right] \qquad ...(2)$$

Variable h is decreased from maximum to minimum, b_1, b_2 and β modify values rp while rg measure the Cartesian distance between the historical best position and their population and in swarms. Cartesian distance is the second distance array standard.

$$\left\| x_i - x_j \right\| = \sum_{s=1}^{n} (x_{is} - x_{js})^2 \quad ...(3)$$

Otherwise, if $f(x_{hi}) < f(x_i)$ implies male mayflies would refresh its speeds from the ongoing one with an irregular D (dance coefficient):

$$v_i(n+1) = h.v_i(n) + D.r_1 \quad ...(4)$$

r_1 is a random value between [-1, 1]. In MMO, the best female and male mayfly is the first mate, the second best is the next, etc. Hence for the i^{th} female mayfly, if $f(y_i) < f(x_i)$:

$$v_i(n+1) = h.v_i(n) + b_3\, e^{-\beta\gamma^2_{mf}} \left[x_i(n) - y_i(n)\right] \quad ...(5)$$

Let b_3 be another velocity-changing constant. r_m addresses the Cartesian distance between them. Otherwise, if $f(yi) < f(xi)$, female mayflies would refresh their speeds from the ongoing one with other random dance fz,

$$v_i(n+1) = h.v_i(n) + f_z.r_2 \quad ...(6)$$

The random number r_2 is in uniform distribution in domain interval [-1, 1]. Their offspring randomly from their parents as follows:

$$OS_1 = N \times male(1 - N) \times female \quad ...(7)$$

$$OS_2 = N \times male(1 - N) \times male \quad ...(8)$$

The random number N is in Gauss distribution. As indicated by conditions (2) and (5), speeds are refreshed from weighted current speeds to other weighted distances and historical best directions, global best candidates or their mates. In more detail, a portion of the weighted distances are displayed as follows:

$$v_p = b_i\, e^{-\beta j^2} \left(q_j - q_i\right) \quad ...(9)$$

Clearly, r_j would be bigger assuming that the distance between j^{th} and i^{th} individuals expanded. But negative outstanding capacity root loads for the distance will be smaller of all things considered. This really intends that if the distance among q^j and q^i is expanded, the loads will decrease, and then the mixed velocity v_p would be then diminished. Then again, if the distance among q^j and q^i is diminished, the weights considered. Whenever the people are far apart, they ought to refresh their speeds at higher rates and when they are closer, and the speeds ought to be refreshed at smaller rates. The weighted distances can be optimized as follows:

$$v_p = b_i\, e^{\frac{-\beta}{r_j}} \left(q_j - q_i\right) \quad ...(10)$$

The pseudocode for data pre-processing using proposed MMO algorithm is described in Algorithm 1.

3.2. Data Normalization Using NLP Pipeline

NLP tasks include word tokenization, parsing, information extraction, recognizing named entity, word sense disambiguation, tagging of parts of speech, stop word analysis, word stemming and lemmatization.

The input from MMO is raw words in sentences which were tokenized by breaking the string into words. Stemming finds the text's word roots by removing the suffix to determine its meaning hence giving the word's basic meaning. For accurate sentiment analysis, we only retained stem words with more than two letters. Then the structure of the words are changed to its dictionary form by filtering affixes or modifying vowels to get lemmas which can be used to reach WordNet for extraction of meaning and sense number for a higher sentiment score. Stop words like "and," "the," "am," and "is" don't have much of an emotional impact and don't change the sentiment score when they're used with lexical resources. Keeping some stop words in the

Algorithm 1: Data Preprocessing Using MMO Algorithm	
Input: Tweets, number of population, threshold condition	
Output: Pre-processed tweets	
1	Objective function f(n), n=(n_1,..., n_d)^T
2	Define the female mayfly individual velocities v_{fi} and y_i(1=1,2,….L)
	Define the male mayfly individual velocities v_{mi} and n_i (i=1, 2 ,….K)
3	Evaluate solutions
4	Search global best g-best
5	Do While stopping criteria are not met
6	Update female, male velocities and solution
7	Evaluate solutions
8	Select the Mayflies
9	Mate the mayflies
10	Evaluate offspring
11	Separate female and male offspring randomly
12	Replace best solution with poor ones
13	Update p-best, g-best
14	End while
15	End

tweet makes it possible to classify emotions correctly. Tweets were checked for words like "I," "am", "with", "and", "it's", "each", "other", and "with". The last task in the NLP pipeline involved POS tagging which translates POS (like nouns, objects, subjects, adjectives, and verbs) into words, looks at the structure of a sentence, and creates word sense disambiguation.

3.3. Extracting Emotion Words Using WordNet and SentiWordNet

Sentiment Analysis examines text for views, sentiments, and emotions. In this section, we discuss the working process of emotional words extraction and sentimental score computation. WordNet lexicon gives semantic meaning of the lemma, expressions, and information about the word's context for figuring out its sentiment polarity, which transmits the emotional content of the word. SentiWordNet gives the extracted opinionated term a score. It contains opinion information derived from WordNet's database, while every term is awarded numerical scores including sentiment value and gloss. WordNet runs SentiWordNet. It measures positive, negative, and neutrality for sentiment analysis.

3.4. Computing Sentiment Polarity Score Using an Improved Horse Herd Optimization Algorithm (IHHO)

SentiWordNet presents multiple methods to assess the emotion of tweets. When the negative score is lower than positive score, the tweet *t* is positive (1); else, it is neutral (0). An improved HHO algorithm is developed for optimal feature selection by reducing dimensionality. Improvement is achieved by examining the horses' natural behavior. Horses' most prevalent behaviors include grazing, hierarchy, sociability, imitation, defense, and wandering. This strategy is inspired by horse behavior at different ages. Horses are moved according to,

$$p_M^{iter,age} = vel_M^{iter,age} + p_M^{(iter-1),age}, \qquad age = \alpha, \beta, \gamma\,\delta \qquad ...(11)$$

where,

- $p_M^{iter,age}$ implies the M-th horse position.
- *age* implies the every horse' range.
- *iter* signifies the current number of iterations.
- $vel_M^{iter,age}$ illustrates horse's vector velocity.

Equations can be depicted as horse motion vectors during every cycle of the method.

$$vel_M^{iter,\alpha} = gra_M^{iter,\alpha} + defmec_M^{iter,\alpha} \quad ...(12)$$

$$vel_M^{iter,\beta} = gra_M^{iter,\beta} + h_m^{iter,\beta} + soc_m^{iter,\beta} + defmec_M^{iter,\beta} \quad ...(13)$$

$$vel_M^{iter,\gamma} = gra_M^{iter,\gamma} + h_m^{iter,\gamma} + soc_m^{iter,\gamma} + imt_m^{iter,\gamma} + ro_m^{iter,\gamma} + defmec_M^{iter,\beta} \quad ...(14)$$

$$vel_M^{iter,\delta} = gra_M^{iter,\delta} + imt_m^{iter,\delta} + ro_m^{iter,\delta} \quad ...(15)$$

Modeling each horse's grazing area using IHHO. Horses graze from birth until death.

$$gra_M^{iter,age} = giter(low + R * upp)(p_M^{(iter-1)}), \quad age = \alpha, \beta, \gamma\, \delta \quad ...(16)$$

$$g_M^{iter,age} = w_g \times (g_M^{(iter-1)}), \quad ...(17)$$

Here, $gra_M^{iter,age}$ implies j^{th} horse motion parameter and signifies the grazing ability by related horse. Each repetition, grazing linearly decreases. "low" and "upp" are the lowest and higher grazing space borders, respectively. At ages 5-15, horses obey the rule of hierarchy, as per research.

$$h_M^{iter,age} = H_M^{iter,age}(p_{LBH}^{(iter-1)} - p_M^{(iter-1)}), \quad ...(18)$$

$$h_M^{iter,age} = H_M^{(-1+iter),age} \times w_g \quad ...(19)$$

Here, $h_M^{iter,age}$ implies the best horse's location. The value $p_{LBH}^{(iter-1)}$ implies the best horse' position. Horses require relationships and can live with other animals. Owing to their particular social features, horses frequently fight. Some horses like being with cattle and sheep, but they don't like being alone. Equations reveal that horses aged 5 to 15 prefer to stay with the herd.

$$Soc_M^{iter,age} = Soc_M^{iter,age}\left[\left(\frac{1}{n}\sum_{i=1}^{n} p_i^{(-1+ier)}\right)\right] age = \beta, \gamma \quad ...(20)$$

$$soc_M^{iter,age} = Soc_M^{(-1+iter),age} \times w_{soc} \quad ...(21)$$

where,

- $Soc_M^{iter,age}$ implies j^{th} horse's social motion vector.
- $soc_M^{iter,age}$ implies horse movement towards group j^{th}.
- *iter*, Every cycle minimizes iteration by w_s.
- the total number of horses be n
- Every horse' age range is represented by *age*. The coefficient t for β and γ horses are calculated from these parameters.

In the present strategy, horse imitation is factor j.

$$im_M^{iter,age} = im_M^{iter,age}\left[\left(\frac{1}{Pn}\sum_{i=1}^{Pn} P_i^{(-1+iter)}\right) - P^{(-1+iter)}\right] \; age = \gamma \quad ...(22)$$

$$im_M^{iter,age} = im_M^{iter,age} \times w_{im} \quad ...(23)$$

Where,

- $im_M^{iter,age}$ expresses the jth horse's motion vector around the best horse in Q position.
- $im_M^{iter,age}$ signifies horse's movement in the i^{th} iteration. This parameter w_{im} is decreased each cycle.
- Qn is the number of best-placed horses, and p signifies 10% of the selection.
- w_{im} implies reduction factor per cycle for *iter*.

A negative coefficient symbolizes the horse's defense mechanism, which keeps it secure.

$$defmec_M^{iter,age} = defmec_M^{iter,age}\left[\left(\frac{1}{Qn}\sum_{i=1}^{Qn} p_i^{(-1+iter)}\right) - o^{(-1+iter)}\right] \; age = \alpha, \beta, \gamma \quad ...(24)$$

$$defmec_M^{iter,age} = defmec_M^{(-1+iter),age} \times w_{defmec} \quad ...(25)$$

- $defmec_M^{iter,age}$ describe the jth horse's escape vector, depending on the worst P position.
- Qn represents the worst-placed horses, where p is 20% of the total.
- w_{defmec} implies $iIter$'s cycle-by-cycle reduction factor.

This technique imitates this behavior using the random factor r. Young horses rarely roam, and it disappears with age.

$$ro_M^{iter,age} = ro_M^{iter,age} \partial P^{(-1+iter)} age = \gamma, \delta \quad ...(26)$$

$$ro_M^{iter,age} = ro_M^{(-1+iter),age} \times w_{ro} \quad ...(27)$$

Here, random i^{th} horse velocity vector for local search and escape from local minima is implied y $ro_M^{iter,age}$: Represent $ro_M^{iter,age}$ per cycle as reduction factor by w_{r0}. The Algorithm 2 describes the working function of sentiment polarity score computation using IHHO algorithm.

Algorithm 2: Compute Sentiment Polarity Score Using IHHO Algorithm	
Input: Multiple features	
Output: Optimal best features	
1	Initialize the best population
2	Define horse movement $p_M^{iter,age} = vel_M^{iter,age} + p_M^{(iter-1),age}$, $age = \alpha, \beta, \gamma\ \delta$
3	While Do apply the $age = \alpha, \beta, \gamma\ \delta$
4	If j = 0, i = 1
5	Each technique cycle has different-aged horse vectors.
6	Define the law of hierarchy $h_M^{iter,age} = H_M^{iter,age}(p_{LBH}^{(iter-1)} - p_M^{(iter-1)})$,
7	Define imitate rule $im_M^{iter,age} = im_M^{iter,age} \times w_{im}$
8	Define optimal fitness using $ro_M^{iter,age} = ro_M^{iter,age} \partial P^{(-1+iter)} age = \gamma, \delta$
9	Update the final value of IHHO
10	End

3.5. Classification of Tweets Using a Hybrid Deep Belief Artificial Neural Network (DB-ANN)

A hybrid DB-ANN is proposed for detecting emotion and classification of tweets. Although ANN has capabilities for decision modeling, limitations may occur when available time series for substantially non-stationary with seasonal changes or ANN training lack sufficient data samples. In such cases, ANN's main framework and understanding is not well suited for such data processing tasks. Dynamic interaction of emotions are better handles by either artificial intelligent systems or DB-ANN models that include artificial emotion. Biologically, animal mood and emotion owing to hormone gland activity may change neuro-physiological response, sometimes by delivering various behaviors for a similar task. Similarly, DB-ANN has a feedback loop between the neurological and hormonal models, which increases the network's learning capabilities. The explicit equation for DB-ANN output is,

$$\hat{x}_i = F_i[\sum_{g=1}^{M} W_{ig} \times F_g(\sum_{j=1}^{N} W_{jg} y_j + W_{ga}) + W_{ia}] \quad ...(28)$$

where W is the applied weight via the neuron; a is the bias; output neuron layer is i, the input neuron layer is j, the hidden layer neuron layer is g. F_g, F_j implies hidden and output layers' activation function; M, N, and y_j implies number of hidden neurons, number of input and input layer variable; x , $\hat{x}_i$ signifies observed and output neuron' computed values. The j-th neuron output in DB-ANN classifier with three hormonal glands of G_b, G_a and G_c can be computed as:

$$x_i = (\lambda_j + \sum_g \sigma_{jg} G_g \times F\ (\sum_j [(\beta_j + \sum_g \zeta_{j,g} G_g) + (\theta_{j,i} + \sum_g \phi_{j,i,K} G_g) Y_{j,i} + (\alpha_j + \sum_g \chi_{j,g} G_g)] \quad ...(29)$$

While DB-ANN classifier's overall hormone value is:

$$G_g = \sum_j G_{j,g} \qquad (g = a, b, c) \quad ...(30)$$

$$G_{j,g} = glandity_{j,g} \times X_j \quad ...(31)$$

In which $glandity_{j,g}$ calibration parameter generates hormonal level of every gland. Various approaches used to initialize every hormone's value G_g (e.g., mean of sample input parameters). The output neuron's error value (D) is used to change the hidden layer's (W_{ig}) and bias (W_{ia})as:

$$W_{ig}(New) = W_{ig}(Old) + \eta.\Delta.XG_g + \alpha W_{ig}(Old) \quad ...(32)$$

$$W_{ia}(New) = W_{ia}(Old) + \eta.\Delta + \alpha.[\delta W_{ia}(Old)] \quad ...(33)$$

where, YG_g signifies g-th hidden neuron output and $d\,\alpha W_{ia}(Old)$ and $\alpha W_{ig}(Old)$ implies bias and weight values' last alterations. Also update the emotional weight (W_{im}) by:

$$W_{iM}(New) = W_{iM}(Old) + \mu.\Delta.X_{avg} + K.[\delta W_{iM}(Old)] \quad ...(34)$$

where, Y_{avg} signifies every epoch's mean input pattern mean and $\alpha W_{im}(Old)$ implies emotional weight transfer.

$$\mu = X_{avg} + \Delta \quad ...(35)$$

$$K = \mu_0 - \mu \quad ...(36)$$

where, μ_0 implies end-of-first-iteration anxiety factor.

$$dc = 1 - \frac{\sum_{j=1}^{n} (P_j - \hat{x}_j)^2}{\sum_{j=1}^{n} (P_j - \overline{P_j})^2} \quad ...(37)$$

$$rmse = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(P_j - \hat{x}_j)^2} \quad ...(38)$$

where dc, RMSE, P_j , n, $\overline{P_j}$ and $\hat{x}_j$ implies determination coefficient, observed data, number of observations, observed data' mean and estimated values used to evaluate model's capturing of runoff peak values as:

$$dc_{peak} = 1 - \frac{\sum_{j=1}^{n_q}(P_{qc_j} - P_{qo_j})^2}{\sum_{j=1}^{n_q}(P_{qo_j} - \overline{P_{qo}})} \quad ...(39)$$

Multi-category Heidke Skill Score (HSS) is computed to evaluate DB-ANN classifier performance in high and low flow regimes.

$$HSS = \frac{\frac{1}{n}\sum_{j=1}^{a}T(\hat{x}_j, P_j) - \frac{1}{n^2}\sum_{j=1}^{a}T(P_j)\times T(\hat{x}_j)}{1 - \frac{1}{n^2}\sum_{j=1}^{a}T(P_j)\times T(\hat{x}_j)} \quad ...(40)$$

Both observed (P_j) and predicted ($\hat{x}_j$) time intervals are junked at same time series to calculate HSS. Therefore, $T(\hat{x}_j)$ (total number of forecasts) and total number of observations $T(P_j)$ in class j are the same. HSS evaluates the fraction of valid predictions after excluding random chance. In the next section, we present the implementation of the proposed methodology as described in the section.

4. Implementation, Results and Applications

In this section, we present the implementation of the methodology presented in this paper. The proposed methodology is executed using PYTHON and NLP toolkit (NLTK). The methodology is validated using standard benchmark datasets such as Crowdflower-1, Crowdflower-2 and Kaggle twitter. The simulation results of proposed WordNet + deep learning is compared to standard benchmark algorithms such as KNN, SVM, DT, NB, LR and RF using the parameters: Recall, F-measure. precision and accuracy.

Table 2: Dataset Description

Datasets	Description	Number of Tweets		
		Training	Testing	Validation
Crowdflower-1	Clean	2800	300	300
	Offensive	2600	200	200
	Hateful	2600	200	200
Crowdflower-2	Clean	2800	300	300
	Offensive	2600	200	200
	Hateful	2600	200	200
Kaggle	Clean	2800	300	300
	Offensive	2600	200	200
	Hateful	2600	200	200
Total Tweets		24,000	2,100	2,100

4.1. Dataset Description

Crowdflower-1 comprised about 14,000 tweets carefully classified as Hateful, Offensive, or Clean by annotated by three people. Crowdflower-2 was tagged as Offensive, Hateful or Neither. The clean class represents the last. The third dataset was Kaggle dataset which classified tweets as sexism, racism, or neither. The first two kinds of hate speech are included in the class hateful. In this experiment, the total 25,020 tweets are split into three subsets as training, testing and validation set with 21,000, 2100 and 2100 tweets respectively. Table 2 describes the dataset details used in this experiment.

4.2. Results on Crowdflower-1 Dataset

Table 3 describes the comparative analysis of our proposed and existing sentimental analysis for the emotional twitter data detection and classification for Crowdflower-1 dataset.

Figure 2 illustrates the accuracy comparison of our proposed and existing scheme. The accuracy of our proposed algorithm is 17.714%, 20.391%, 23.068%, 25.745%, 28.422% and 31.099% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 3 depicts the precision comparison of our proposed and existing scheme. The precision of the proposed algorithm is 17.757%, 20.440% , 23.124%, 25.87%, 28.49% and 31.174% efficient than KNN, SVM, DT, NB, LR and RF schemes.

Table 3: Comparative Analysis for Crowdflower-1 Dataset

Sentimental Analysis Scheme	%			
	Accuracy	Precision	Recall	F-measure
Random forest (RF)	65.89	65.66	67.055	66.35
Linear regression (LR)	68.45	68.22	69.615	68.91
Navie Bayes (NB)	71.01	70.78	72.175	71.471
Decision tree (DT)	73.57	73.34	74.735	74.031
Support vector machine (SVM)	76.13	75.9	77.295	76.591
K-nearest neighbor (k-NN)	78.69	78.46	87.045	82.53
WordNet+deep learning	95.63	95.4	95.63	95.515

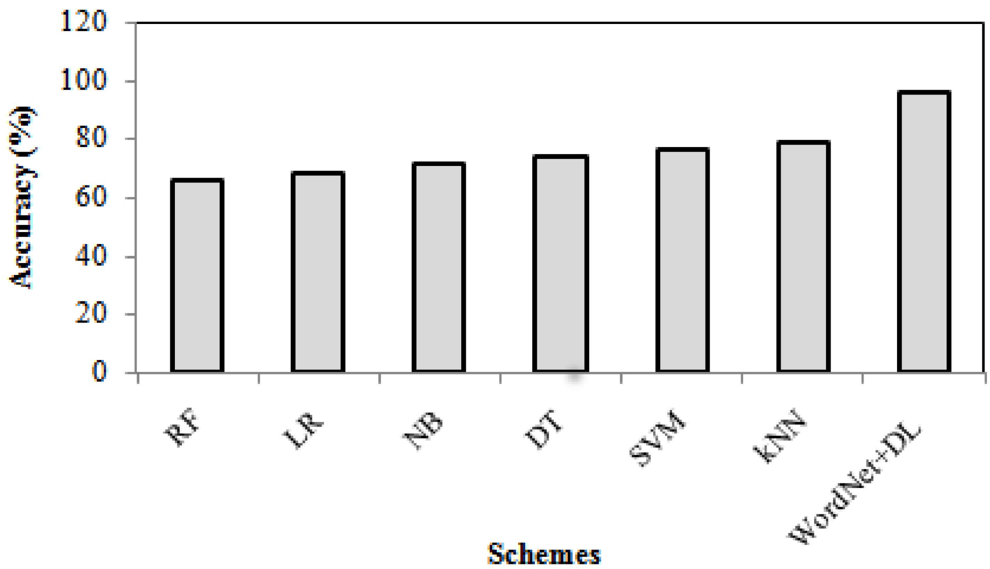

Figure 2: Accuracy Comparison for Crowdflower-1 Dataset

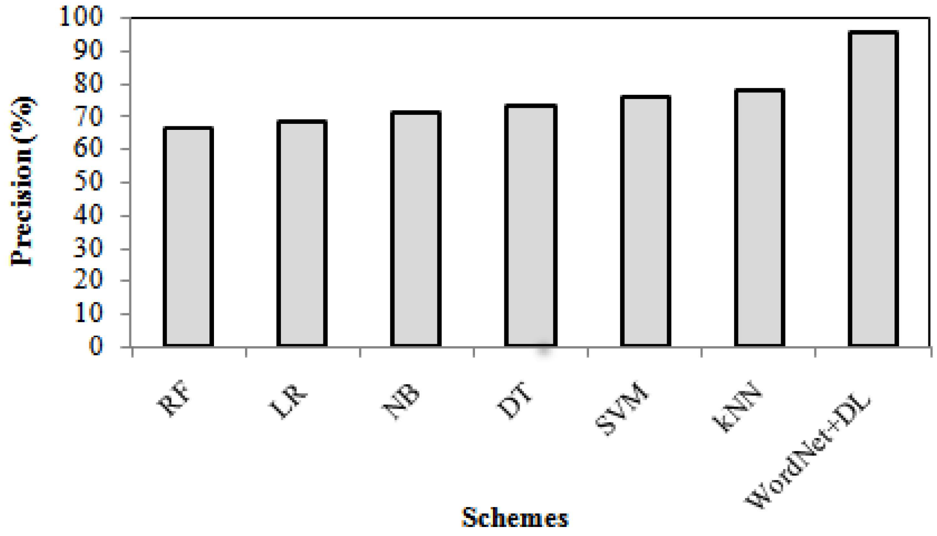

Figure 3: Precision Comparison for Crowdflower-1 Dataset

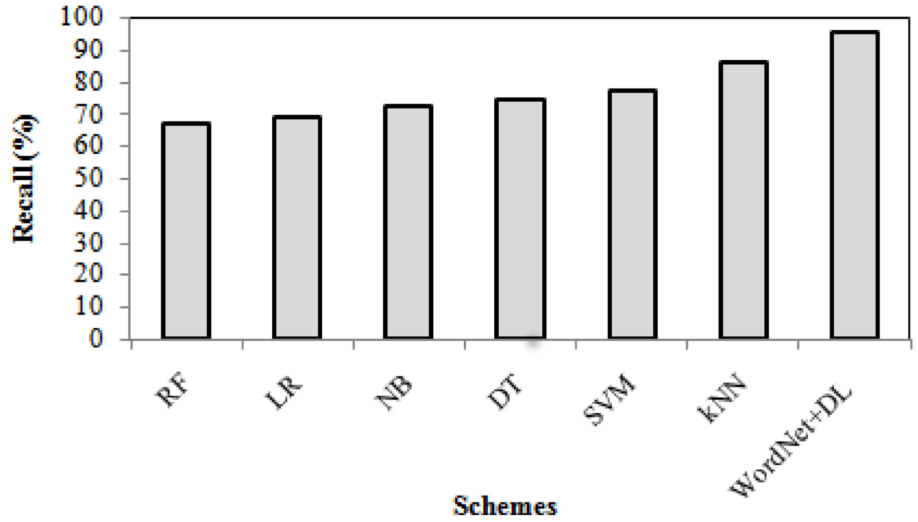

Figure 4: Recall Comparison for Crowdflower-1 Dataset

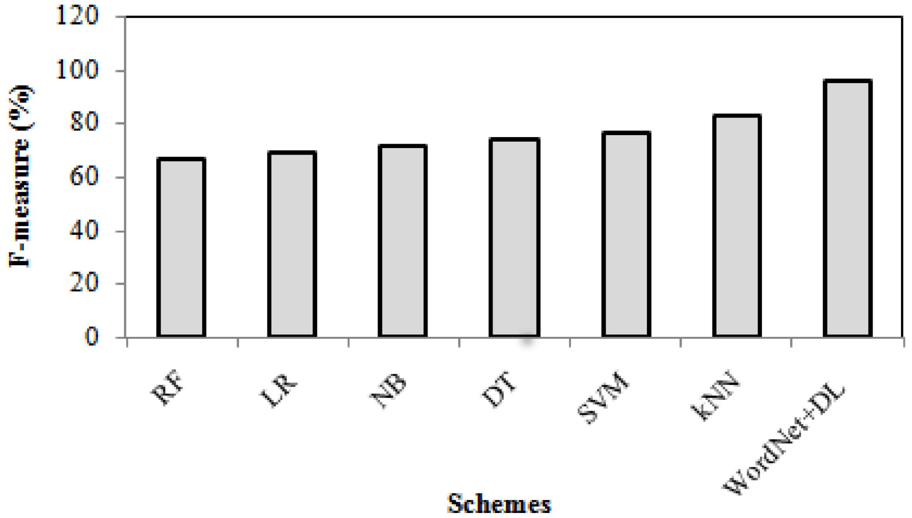

Figure 5: F-measure Comparison for Crowdflower-1 Dataset

Figure 4 illustrates the Recall comparison of our proposed and existing scheme. The recall of our proposed algorithm is 8.977%, 19.173%, 21.850%, 24.527%, 27.204%, 29.881% efficient than existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 5 compares the F-measure of our proposed and existing scheme. Our proposed algorithms' F-measure is 13.595% , 19.82%, 22.49%, 25.173%, 27.854% and 30.534% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

4.3. Results on Crowdflower-2 Dataset

Table 4 describes the comparative analysis of our proposed and existing sentimental analysis for the emotional twitter data detection and classification for Crowdflower-2 dataset.

Figure 6 depicts the accuracy comparison of our proposed and existing scheme. The accuracy of our proposed scheme is 9.113%, 11.756%, 14.398%, 17.04%, 19.682% and 22.34%, efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 7 displays the precision comparison of our proposed and existing scheme. The precision of our proposed scheme is 9.135%, 11.784%, 14.432%, 17.08%, 19.729% and 22.377%, efficient than the existing KNN, SVM, DT, NB, LR and RF schemes

Table 4: Comparative Analysis for Crowdflower-2 Dataset

Sentimental Analysis Scheme	%			
	Accuracy	Precision	Recall	F-measure
Random forest (RF)	75.26	75.03	76.425	75.721
Linear regression (LR)	77.82	77.59	78.985	78.281
Navie Bayes (NB)	80.38	80.15	81.545	80.841
Decision tree (DT)	82.94	82.71	84.105	83.402
Support vector machine (SVM)	85.5	85.27	86.665	85.962
K-nearest neighbor (k-NN)	88.06	87.83	92.36	90.038
WordNet+deep learning	96.89	96.66	96.89	96.775

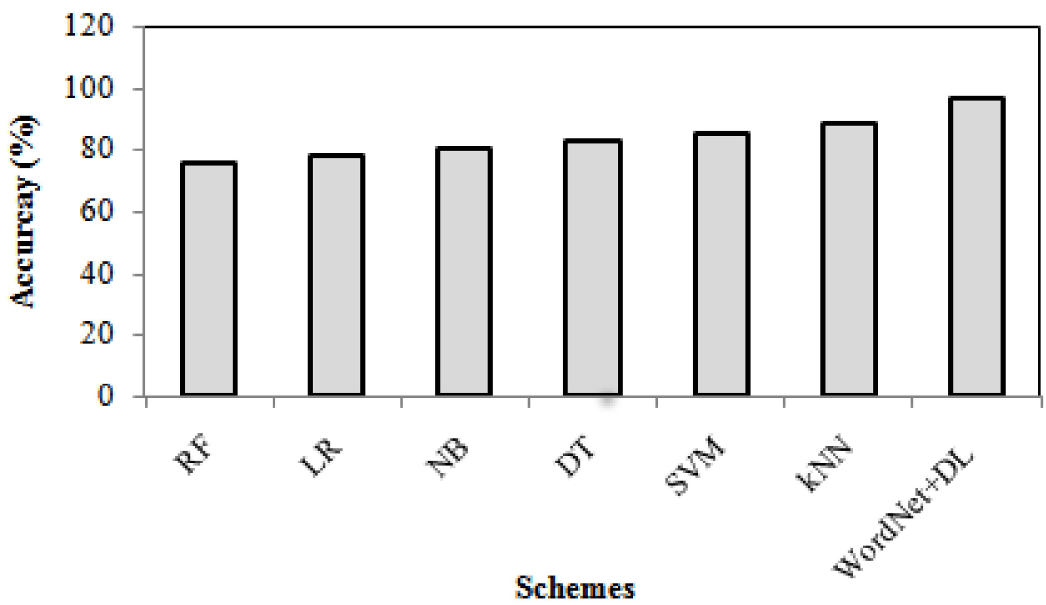

Figure 6: Accuracy Comparison for Crowdflower-2 Dataset

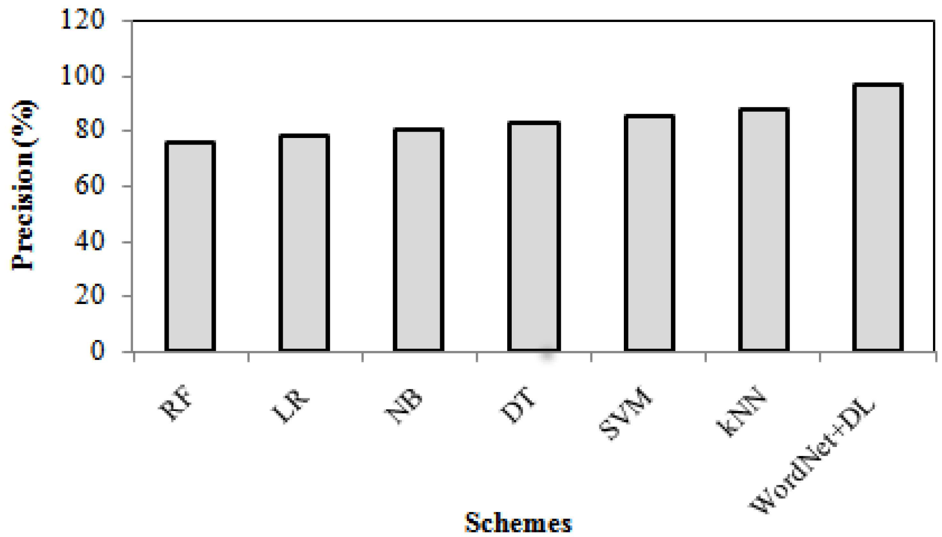

Figure 7: Precision Comparison for Crowdflower-2 Dataset

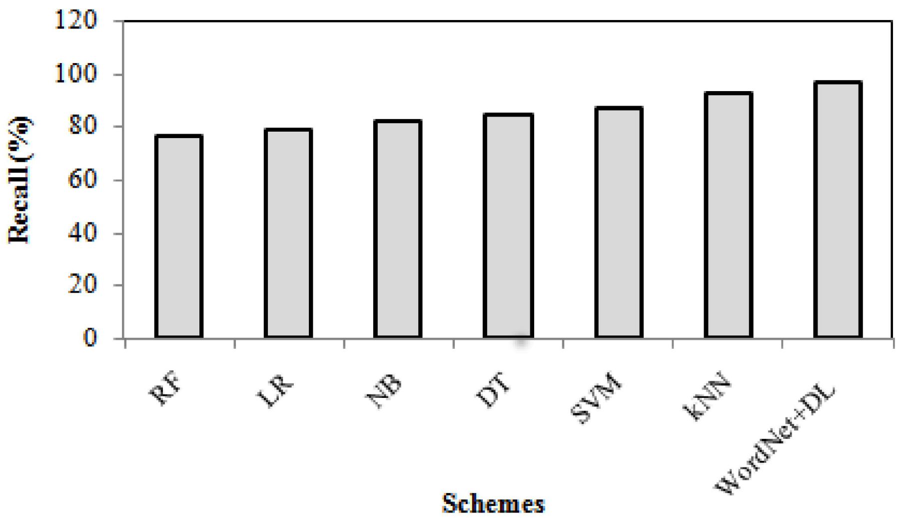

Figure 8: Recall Comparison for Crowdflower-2 Dataset

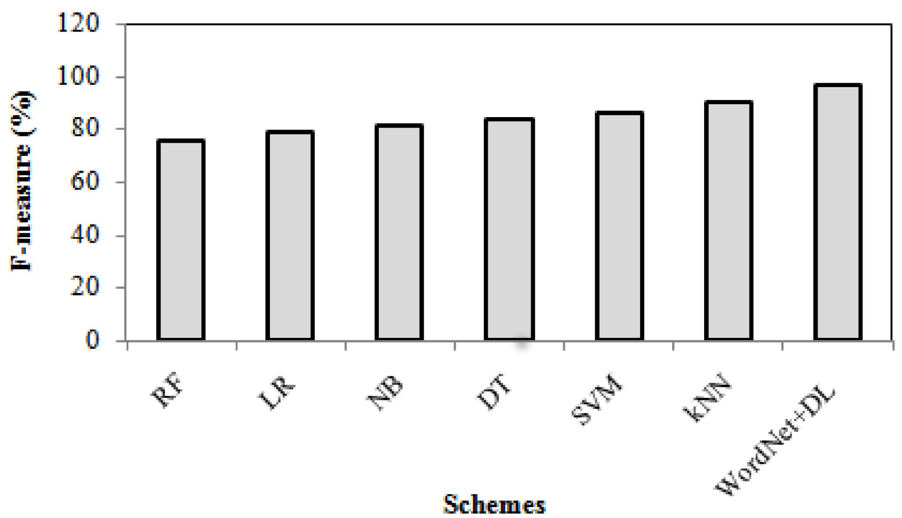

Figure 9: F-measure Comparison for Crowdflower-2 Dataset

The recall comparison of proposed and existing scheme is demonstrated by Figure 8. The recall of our proposed scheme is 4.675%, 10.553%. 13.195%, 15.838%, 18.48% and 21.122% efficient than existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 9 demonstrates the F-measure comparison of our proposed and existing scheme. The F-measure of our proposed scheme is 6.961%, 11.173%, 13.819%, 16.464%, 19.11%, 21.755% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

4.4. Results on Kaggle Twitter Dataset

Table 5 describes the comparative analysis of our proposed and existing sentimental analysis for emotional twitter data detection and classification for Kaggle twitter dataset.

Figure 10 demonstrates the accuracy comparison of our proposed and existing scheme. The accuracy of our proposed scheme is 6.353%, 8.976%, 11.6%, 14.223%, 16.846% and 19.469% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 11 represents the precision comparison of proposed and existing scheme. The precision of our proposed scheme is 6.368%, 8.998%, 11.627%, 14.256%, 16.886% and 19.515% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

Table 5: Comparative Analysis for Kaggle Twitter Dataset

Sentimental Analysis Scheme	%			
	Accuracy	Precision	Recall	F-measure
Random forest (RF)	78.59	78.36	79.755	79.051
Linear regression (LR)	81.15	80.92	82.315	81.612
Navie Bayes (NB)	83.71	83.48	84.875	84.172
Decision tree (DT)	86.27	86.04	87.435	86.732
Support vector machine (SVM)	88.83	88.6	89.995	89.292
K-nearest neighbor (k-NN)	91.39	91.16	94.375	92.74
WordNet+deep learning	97.59	97.36	97.59	97.475

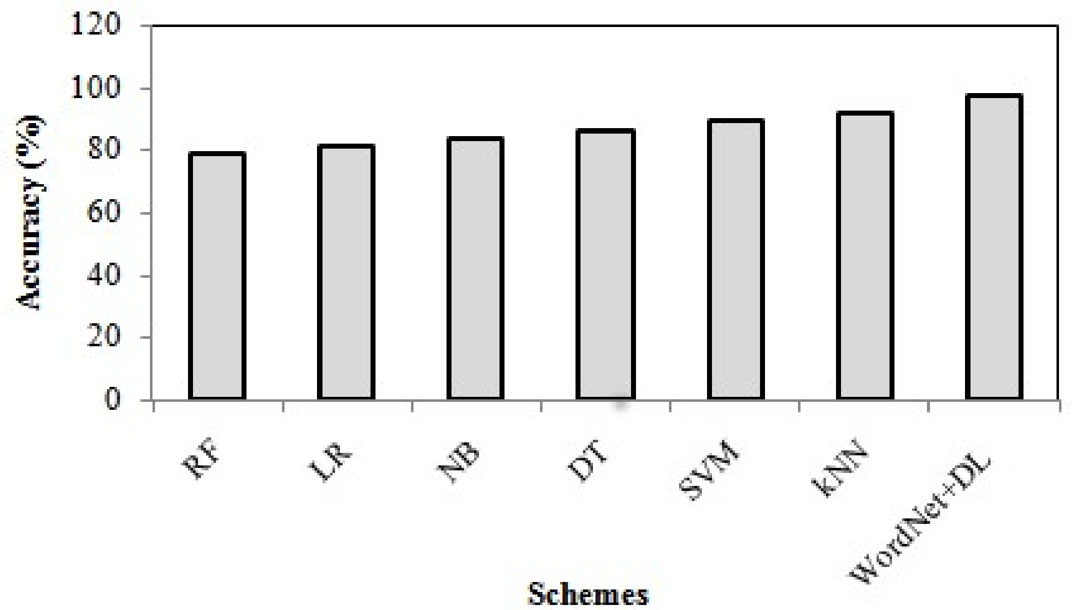

Figure 10: Accuracy Comparison for Kaggle Twitter Dataset

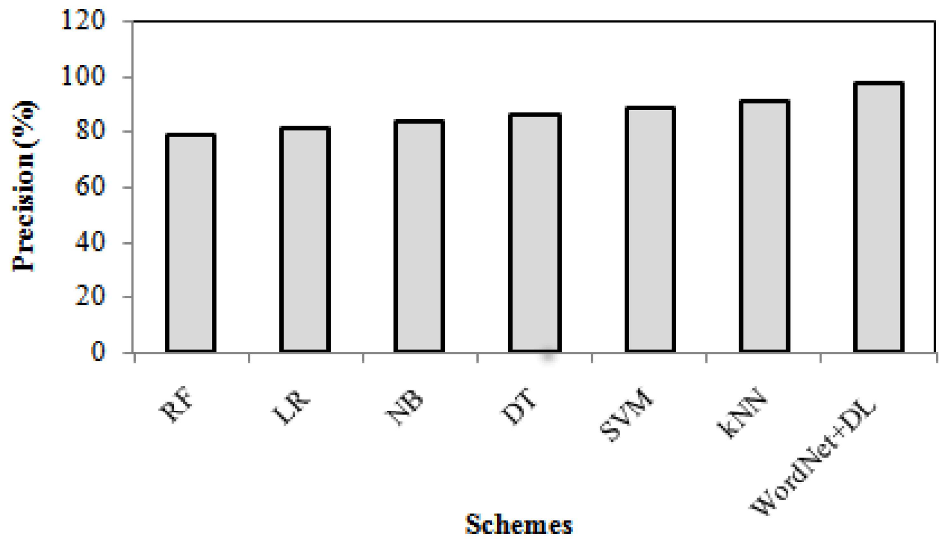

Figure 11: Precision Comparison for Kaggle Twitter Dataset

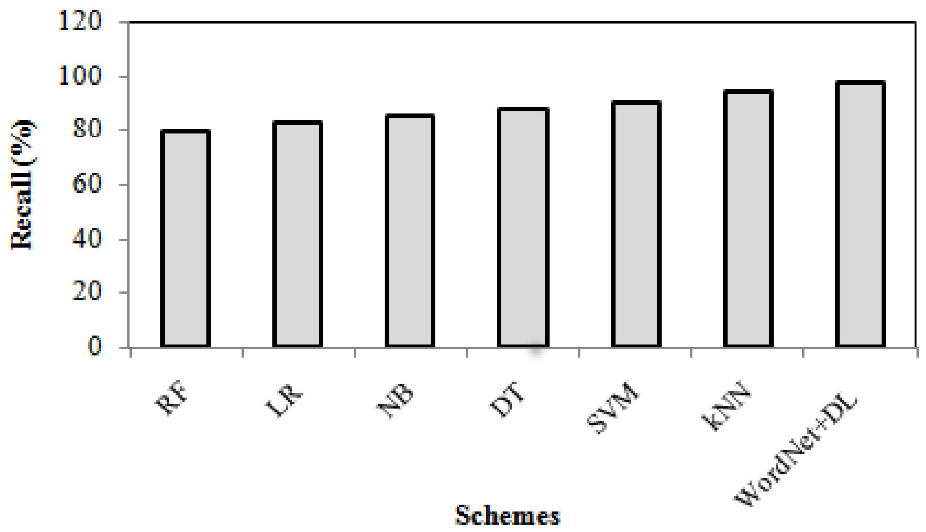

Figure 12: Recall Comparison for Kaggle Twitter Dataset

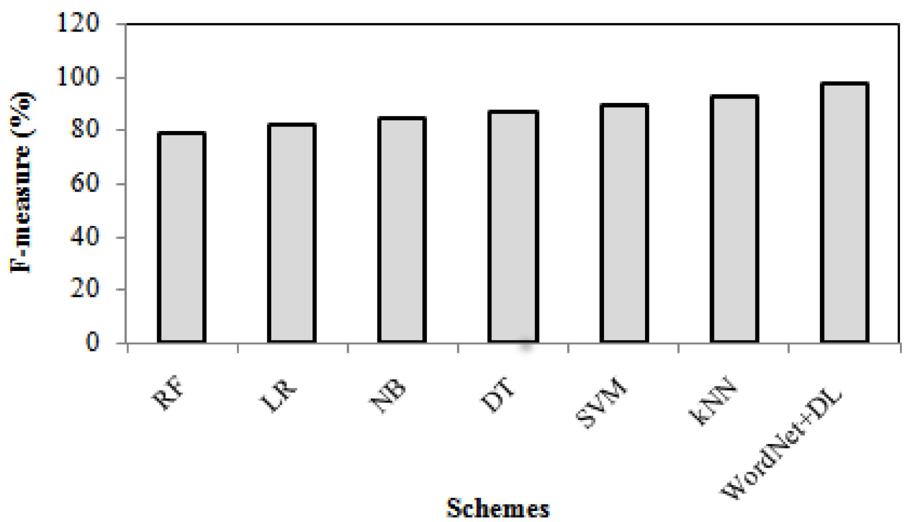

Figure 13: F-measure Comparison for Kaggle Twitters Dataset

Figure 12 illustrates the recall comparison of proposed and existing scheme. Recall of our proposed scheme is 18.275%, 15.652%, 13.029%, 10.406%, 7.783% and 3.294% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes.

Figure 13 illustrates the F-measure comparison of our proposed and existing scheme. The F-measure of our proposed scheme is 18.901%, 16.274%, 13.648%, 11.021%, 8.395% and 4.858% efficient than the existing KNN, SVM, DT, NB, LR and RF schemes

4.5. Applications of Detected Emotions

There are several applications of the proposed technique for detecting and classifying tweets presented in this paper. Here we list three major applications (that are within the context of emotions detection) as this is the theme of this paper, as follows:

1. Hate speech detection
2. Crime Alert in Real Time
3. Crime Reports
4. Hotspot identification and Map Annotation

5. Conclusion and Future Work

We have proposed a novel sentiment analysis for emotion twitter data using optimal WordNet-based emotional word extraction and hybrid DB-ANN learning classifier. MMO algorithm is used for data pre-processing which filtered unwanted noises from the raw data. Then, we normalized the data using NLP pipeline. IHHO algorithm was used for the emotional words extraction from the tweets with the help of WordNet and to compute sentiment polarity using the SentiWordNet dictionary. A Hybrid DB-ANN was then used for Twitter data sentiment analysis to ensure the better detection rate. The proposed technique was validated against standard benchmark datasets and results compared with industry standard algorithms using four measurements: accuracy, precision, Recall, and F-measure. Our proposed learning scheme achieved the best accuracy of 95.63%, 96.89% and 97.59% for Crowdflower-1, Crowdflower-2, and Kaggle Twitter datasets respectively. For future work, we will work at further improving MMO to incorporate sentence splitter for NLP pipeline.

Author Contribution

1. Stephen Obare: Conceptualization, Writing, Methodology, Implementation, Results and Analysis, Original draft preparation, final writing.
2. Abejide Ade-Ibijola.: Reviewing, Editing, Supervision and Advisory.
3. Kennedy Ogada: Reviewing the paper.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of Interest

The authors declare no conflict of interest.

References

Akilandeswari, J. and Jothi, G. (2018). Sentiment Classification of Tweets with Non-language Features. *Procedia Computer Science,* 143, 426-433.

AL-Sharuee, M.T., Liu, F. and Pratama, M. (2018). Sentiment Analysis: An Automatic Contextual Analysis and Ensemble Clustering Approach and Comparison. *Data & Knowledge Engineering*, 115, 194-213.

Asgarian, E., Kahani, M. and Sharifi, S. (2018). The Impact Of Sentiment Features on the Sentiment Polarity Classification in Persian Reviews. *Cognitive Computation*, 10(1), 117-135.

Ayo, F.E., Folorunso, O., Ibharalu, F.T., Osinuga, I.A. and Abayomi-Alli, A. (2021). A Probabilistic Clustering Model for Hate Speech Classification in Twitter. *Expert Systems with Applications*, 173, 114762.

Baydogan, C. and Alatas, B. (2021). Metaheuristic Ant Lion and Moth Flame Optimization-Based Novel Approach for Automatic Detection of Hate Speech in Online Social Networks. *IEEE Access*, 9, 110047-110062.

Beddiar, D.R., Jahan, M.S. and Oussalah, M. (2021). Data Expansion Using Back Translation and Paraphrasing for Hate Speech Detection. *Online Social Networks and Media,* 24, 100153.

Cruz, R.M., de Sousa, W.V. and Cavalcanti, G.D. (2022). Selecting and Combining Complementary Feature Representations and Classifiers for Hate Speech Detection. arXiv preprint arXiv:2201.06721.

Das, A.K., Al Asif, A., Paul, A. and Hossain, M.N. (2021). Bangla Hate Speech Detection on Social Media Using Attention-based Recurrent Neural Network. *Journal of Intelligent Systems,* 30(1), 578-591.

Dehkharghani, R., Saygin, Y., Yanikoglu, B. and Oflazer, K. (2016). SentiTurkNet: A Turkish Polarity Lexicon for Sentiment Analysis. *Language Resources and Evaluation*, 50(3), 667-685.

Dessì, D. and Diego, R.R. (2021). LexTex: A Framework to Generate Lexicons Using Wordnet Word Senses in Domain Specific Categories. *Journal of Intelligent Information Systems*, 1-24.

Gutiérrez, Y., Vázquez, S. and Montoyo, A. (2017). Spreading Semantic Information by Word Sense Disambiguation. Knowledge-Based Systems, 132, 47-61.

Hung, C. and Chen, S.J. (2016). Word Sense Disambiguation Based Sentiment Lexicons for Sentiment Classification. *Knowledge-Based Systems*, 110, 224-232.

Hung, C. (2017). Word of Mouth Quality Classification Based on Contextual Sentiment Lexicons. *Information Processing & Management,* 53(4), 751-763.

Jha, V., Savitha, R., Shenoy, P.D., Venugopal, K.R. and Sangaiah, A.K. (2018). A Novel Sentiment Aware Dictionary for Multi-domain Sentiment Classification. *Computers & Electrical Engineering*, 69, 585-597.

Kapil, P. and Ekbal, A. (2020). A Deep Neural Network Based Multi-task Learning Approach to Hate Speech Detection. *Knowledge-Based Systems,* 210, 106458.

Khan, F.H., Qamar, U. and Bashir, S. (2016). eSAP: A Decision Support Framework for Enhanced Sentiment Analysis and Polarity Classification. *Information Sciences,* 367, 862-873.

Khan, F.H., Qamar, U. and Bashir, S. (2016). SentiMI: Introducing Point-wise Mutual Information with Sentiwordnet to Improve Sentiment Polarity Detection. *Applied Soft Computing*, 39, 140-153.

Khan, F.H., Qamar, U. and Bashir, S. (2016). SWIMS: Semi-supervised Subjective Feature Weighting and Intelligent Model Selection for Sentiment Analysis. *Knowledge-Based Systems*, 100, 97-111.

Khan, F.H., Qamar, U. and Bashir, S. (2017). A Semi-supervised Approach to Sentiment Analysis Using Revised Sentiment Strength Based on SentiWordNet. *Knowledge and information Systems*, 51(3), 851-872.

Kocoñ, J., Figas, A., Gruza, M., Puchalska, D., Kajdanowicz, T. and Kazienko, P. (2021). Offensive, Aggressive, and Hate Speech Analysis: From Data-centric to Human-centered Approach. *Information Processing & Management*, 58(5), 102643.

Medagoda, N., Shanmuganathan, S. and Whalley, J. (2015), August. Sentiment Lexicon Construction using SentiWordNet 3.0. In 2015 11th International Conference on Natural Computation (ICNC), 802-807, IEEE.

Miranda, E., Aryuni, M., Hariyanto, R. and Surya, E.S. (2019, August). Sentiment Analysis using Sentiwordnet and Machine Learning Approach (Indonesia general election opinion from the twitter content). In 2019 International Conference on Information Management and Technology (ICIMTech), 1, 62-67, *IEEE*.

Mossie, Z. and Wang, J.H. (2020). Vulnerable Community Identification Using Hate Speech Detection on Social Media. *Information Processing & Management,* 57(3), 102087.

Nascimento, F.R., Cavalcanti, G.D. and Da Costa-Abreu, M. (2022). Unintended Bias Evaluation: An Analysis of Hate Speech Detection and Gender Bias Mitigation on Social Media Using Ensemble Learning. *Expert Systems with Applications,* 201, 117032.

Pamungkas, E.W., Basile, V. and Patti, V. (2021). A joint Learning Approach with Knowledge Injection for Zero-shot Cross-lingual Hate Speech Detection. *Information Processing & Management,* 58(4), 102544.

Plaza-del-Arco, F.M., Molina-González, M.D., Urena-López, L.A. and Martín-Valdivia, M. T. (2021). Comparing Pre-trained Language Models for Spanish Hate Speech Detection. *Expert Systems with Applications,* 166, 114120.

Plaza-Del-Arco, F.M., Molina-González, M.D., Ureña-López, L.A. and Martín-Valdivia, M.T. (2021). A Multi-Task Learning Approach to Hate Speech Detection Leveraging Sentiment Analysis. *IEEE Access,* 9, 112478-112489.

Pronoza, E., Panicheva, P., Koltsova, O. and Rosso, P. (2021). Detecting Ethnicity-targeted Hate Speech in Russian Social Media Texts. *Information Processing & Management*, 58(6), 102674.

Qureshi, K.A. and Sabih, M. (2021). Un-compromised credibility: Social Media Based Multi-class Hate Speech Classification for Text. *IEEE Access*, 9, 109465-109477.

Sharma, A., Kabra, A. and Jain, M. (2022). Ceasing Hate with MoH: Hate Speech Detection in Hindi–English Code-switched Language. *Information Processing & Management,* 59(1), 102760.

Watanabe, H., Bouazizi, M. and Ohtsuki, T. (2018). Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection. *IEEE Access,* 6, 13825-13835.

Cite this article as: Stephen Obare, Abejide Ade-Ibijola and Kennedy Ogada (2023). An Optimal WordNet Based Emotional Word Extraction and Hybrid Deep Learning Classifier for Sentiment Analysis. *International Journal of Data Science and Big Data Analytics,* 3(1), 25-44. doi: 10.51483/IJDSBDA.3.1.2023.25-44.