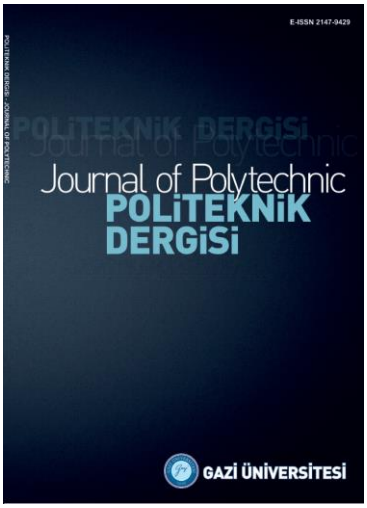

POLİTEKNİK DERGİSİ

JOURNAL of POLYTECHNIC

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)
URL: http://dergipark.org.tr/politeknik

GenSent: Improving Sentiment Analysis Using Genetic Algorithm-Based Ensemble Optimization

GenSent: Genetik Algoritma Tabanlı Topluluk Optimizasyonu Kullanılarak Duygu Analizinin İyileştirilmesi

Yazar(lar) (Author(s)): *Roza Hikmat Hama Aziz*[1], Nazife Dimililer [2]*

ORCID[1]: 0000-0002-8861-4132

ORCID[2]: 0000-0003-2941-8219

To cite to this article: Hama Aziz R. H. and Dimilier N., "GenSent: Improving Sentiment Analysis Using Genetic Algorithm-Based Ensemble Optimization",***Journal of Polytechnic***,*(*): *, (*).

Bu makaleye şu şekilde atıfta bulunabilirsiniz: Hama Aziz R. H. and Dimilier N., "GenSent: Improving Sentiment Analysis Using Genetic Algorithm-Based Ensemble Optimization",***Politeknik Dergisi***, *(*): *, (2025).

Erişim linki (To link to this article): http://dergipark.org.tr/politeknik/archive

DOI: 10.2339/politeknik.1705902

GenSent: Improving Sentiment Analysis Using Genetic Algorithm-Based Ensemble Optimization

Highlights

- *Social media posts are vast, making sentiment analysis challenging and time-consuming.*
- *Strong sentiments like very negative or very positive are also significant, however the majority of studies simply examine fundamental emotions like positive, negative, or neutral.*
- *This paper introduces GenSent, a genetic algorithm-based method for deeper sentiment analysis.*
- *GenSent combines multiple machine learning models automatically to improve the accuracy of sentiment detection.*
- *Tests on well-known datasets show that GenSent works better than other methods and is also faster and more efficient.*

Graphical Abstract

During the ensembling phase, a Genetic Algorithm evolves the optimal classifier ensemble from a large pool of base classifiers, evaluated on well-known sentiment analysis datasets.

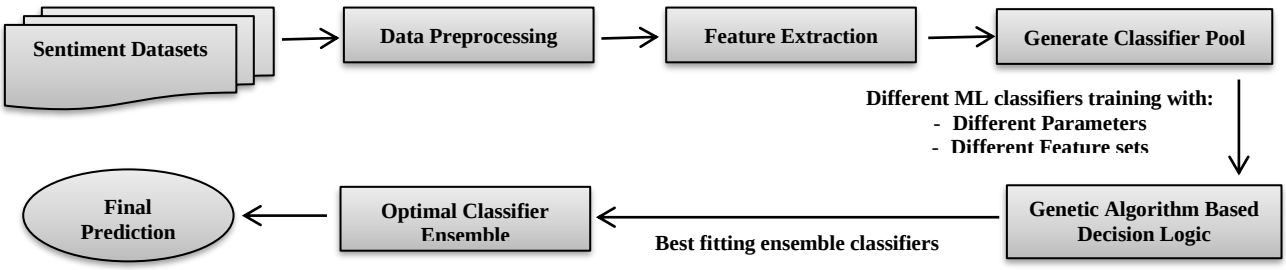

Figure. GenSent Framwork Flowchat

Aim

This study proposes GenSent, a robust sentiment analysis framework using Genetic Algorithms to optimize classifier ensembles for fine-grained sentiment and aspect-level analysis, enhancing accuracy and adaptability.

Design & Methodology

GenSent is a sentiment analysis framework that employs Genetic Algorithms to create optimized ensembles from a pool of 25 classifiers, following four stages: preprocessing, feature extraction, training, and ensemble optimization.

Originality

This paper presents GenSent, a framework that uses Genetic Algorithms to optimize classifier ensembles for fine-grained sentiment detection. It goes beyond basic sentiments by also capturing strong sentiments like very negative and very positive.

Findings

GenSent was evaluated using benchmark datasets and compared with several well-known sentiment analysis methods. The results show that it outperforms existing approaches in accuracy while also reducing computational complexity.

Conclusion

GenSent is a genetic algorithm-based framework that selects optimal classifier subsets for binary, ternary, and fine-grained sentiment analysis, outperforming individual models and existing ensembles.

Declaration of Ethical Standards

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

GenSent: Improving Sentiment Analysis Using Genetic Algorithm-Based Ensemble Optimization

Araştırma Makalesi / Research Article

Roza Hikmat Hama AZİZ*[1], Nazife DİMİLİER[2]

[1]Department of Computer Science, College of Basic Education, University of Sulaimani, Sulaimani 46001, Iraq
[2]School of Computing and Technology, Department of Information Technology, Eastern Mediterranean University (EMU), Famagusta 99450, North Cyprus, Mersin 10 Turkey

(Geliş/Received : 25.05.2025 ; Kabul/Accepted : 28.09.2025 ; Erken Görünüm/Early View : 26.10.2025)

ABSTRACT

Social media platforms are currently the primary medium of all types of communication from personal interactions, and opinion sharing to the dissemination of important international news. However, the ever-increasing amount of user-generated textual information coupled with the dynamic nature of the language, subtle or hidden nuances in expressions used, and contextual dependencies in text, renders timely and accurate sentiment analysis increasingly challenging. Sentiment analysis is an important task in its own right and is also used as the first step of many other classification tasks such as hate speech and misinformation detection. A significant portion of research on sentiment analysis and opinion mining has concentrated on categorizing social media content into three classifications: positive, negative, or neutral. However, despite their importance across numerous practical domains, the classification of extreme opinions, such as highly negative and highly positive sentiments, has only recently gained attention. To address this gap, we propose a framework, GenSent, a novel genetic algorithm-based optimization framework for sentiment classification. Unlike traditional methods that are often tailored to specific datasets, GenSent provides a versatile framework applicable to diverse sentiment analysis tasks from binary, ternary, and fine-grained 5-point scale classification that represents extreme sentiments as well. Through the use of a diverse pool of classifiers including support vector machines, Naïve Bayes, Logistic Regression, Decision Trees, Random Forests, and Stochastic Gradient Descent Algorithms, GenSent effectively builds a robust ensemble without any intervention. The framework is evaluated using binary, ternary, and fine-grained sentiment analysis datasets, namely, SemEval-2017 (Sentiment Analysis in Twitter) task (4A, 4B, and 4C) and Stanford Sentiment Treebank (SST-2 and SST-5). The performance of the proposed framework is compared with other existing well-known methods in the field using the same datasets. Comparative results demonstrate that GenSent outperforms existing methods, achieving significant improvements in sentiment classification across various metrics while reducing the computational complexity.

Keywords: sentiment analysis, machine learning, optimized ensemble classifier, genetic optimization, meta-heuristic algorithms.

GenSent: Genetik Algoritma Tabanlı Topluluk Optimizasyonu Kullanılarak Duygu Analizinin İyileştirilmesi

ÖZ

Sosyal medya platformları, kişisel etkileşimlerden fikir paylaşımına ve önemli uluslararası haberlerin yayılmasına kadar her türlü iletişimin birincil ortamıdır. Bununla birlikte, kullanıcılar tarafından oluşturulan metinsel bilginin sürekli artan miktarı, dilin dinamik yapısı, kullanılan ifadelerdeki ince veya gizli nüanslar ve metindeki bağlamsal bağımlılıklar, zamanında ve doğru duygu analizini giderek daha zorlu hale getirmektedir. Duygu analizi kendi başına önemli bir görevdir ve nefret söylemi ve yanlış bilgi tespiti gibi birçok diğer sınıflandırma görevinin ilk adımı olarak da kullanılır. Duygu analizi ve görüş madenciliği üzerine yapılan araştırmaların önemli bir kısmı, sosyal medya içeriklerini olumlu, olumsuz veya nötr olmak üzere üç sınıfa ayırmaya odaklanmıştır. Ancak, birçok pratik alandaki önemlerine rağmen, son derece olumsuz ve son derece olumlu duygular gibi aşırı görüşlerin sınıflandırılması ancak son zamanlarda ilgi görmeye başlamıştır. Bu boşluğu gidermek için, duygu sınıflandırması için yeni bir genetik algoritma tabanlı optimizasyon çerçevesi olan GenSent adlı bir çerçeve öneriyoruz. Genellikle belirli veri kümelerine göre uyarlanan geleneksel yöntemlerin aksine, GenSent, uç duyguları da temsil eden ikili, üçlü ve ince taneli 5 puanlık ölçek sınıflandırmasından çeşitli duygu analizi görevlerine uygulanabilen çok yönlü bir çerçeve sunar. Destek vektör makineleri, Naïve Bayes, Lojistik Regresyon, Karar Ağaçları, Rastgele Ormanlar ve Stokastik Gradyan İniş Algoritmaları dahil olmak üzere çeşitli sınıflandırıcı havuzlarının kullanımıyla GenSent, herhangi bir müdahale olmaksızın güçlü bir topluluk oluşturur. Çerçeve, ikili, üçlü ve ince taneli duygu analizi veri kümeleri, yani SemEval-2017 (Twitter'da Duygu Analizi) görevi (4A, 4B ve 4C) ve Stanford Duygu Ağaç Bankası (SST-2 ve SST-5) kullanılarak değerlendirilir. Önerilen çerçevenin performansı, aynı veri kümelerini kullanan alanda bilinen diğer mevcut yöntemlerle karşılaştırılır. Karşılaştırmalı sonuçlar, GenSent'in mevcut yöntemlerden daha iyi performans gösterdiğini, hesaplama karmaşıklığını azaltırken çeşitli metriklerde duygu sınıflandırmasında önemli iyileştirmeler sağladığını göstermektedir.

Anahtar Kelimeler: duygu analizi, makine öğrenmesi, optimize edilmiş topluluk sınıflandırıcı, genetik optimizasyon, meta-sezgisel algoritmalar.

1. INTRODUCTION

The production of textual documents has recently increased exponentially in social media and other platforms. Data collected from social media platforms can be used for various purposes including tour consulting services, election forecasts, financial trend

*Sorumlu Yazar (Corresponding Author)
e-posta : roza.hamaaziz@univsul.edu.iq

projections, advertising, and collecting customer feedback or opinions on products and services such as those provided by Amazon [1]. Social media facilitates interactive connections between users and platforms, where people express opinions and establish relations by sharing messages, reviews, comments, and feedback on numerous topics [2]. The data on these interactions and emotions is valuable for consumers as well as advertisers in analyzing public opinion on their topics of interest. However, reading and analyzing all this data can be overwhelming due to its size and the pace at which it is being produced and accumulated. Therefore, there is a need for systems to summarize and process these textual data. Sentiment Analysis is one of the most important application areas targeting this problem.

Sentiment analysis, also known as opinion mining, identifies and predicts people's emotions, feelings, thoughts, and attitudes towards a subject and is a crucial tool in addition to being the first step for many other text mining tasks such as hate speech detection and misinformation detection. For instance, the users of a review-aggregation platform for movies can use the sentiment of other users' comments about that movie to decide whether they want to watch it or not. According to the sentiments of citizens in online comments, governments can adapt their policies [3]. Notably, the emergence and growth of sentiment analysis research coincides with the increasing importance and exponential progress of social media. Pang et al. [6] and Turney [7] carried out seminal work on sentiment analysis to determine the sentiment orientation of phrases or words as positive or negative. Following that, studies were conducted on the linguistic aspects of expressing opinions and views or sentiments in addition to deeper linguistic processing such as negation handling, finer-grained sentiment distinctions, positional information, and the role of context in determining sentiment orientation. [26-29]. Furthermore, Stoyanov and Cardie note that for fine-grained opinion or sentiment analysis, it is essential not only to determine the polarity of sentiment but also the topic category of the sentiment [8]. Later, with the rapid growth of social media, sentiment analysis on Twitter became an important research topic. Unfortunately, the lack of suitable lexicons and databases for training, development, and testing systems hindered research in that direction. Over time, some Twitter resources were developed, but they were limited and proprietary. For instance, the I-sieve corpus [9] was produced just for Spanish and TASS corpus [10] or depended on noisy labels automatically obtained based on hashtags and emotions such as the Hashtag Emotion Corpus [11, 6]. In recent years, this situation changed with the shared task on sentiment analysis on Twitter. The Semantic Evaluation (SemEval) is one of the most important sources of contribution, historically known as the SensEval, which provides public datasets and holds competition on sentiment analysis tasks. Since 2013, this task has been run yearly [14]. The main competition on this task began with SemEval-2013 task 2 [12] and SemEval-2014 task 9 [13] with 2-point scales. Then, in the SemEval-2015 task, sentiment toward a topic was introduced [15], while the SemEval-2016 task added a 5-point scale classification and quantification [16, 17].

Most researchers have attempted to build intelligent automated approaches for improving the performance efficiency and accuracy of analyzing sentiment in tweets utilizing different techniques and architecture. A large number of studies have been conducted to assess the sentiment of tweets and classify them using machine-learning techniques [18, 21, 24, 25]. In a broader sense, the approaches used in sentiment analysis are generally categorized into two distinct groups, namely the supervised and the unsupervised machine learning approaches [18]. In the supervised learning approach, classification models learn from a labeled set of product reviews to construct a model, which then makes predictions on new datasets. Unsupervised learning approaches can either work based on lexicon or machine learning. In these approaches, the classifiers do not always require the labeled data to discriminate the given input text. In unsupervised approaches, only the input text is provided to the classifier; hence, the classifiers do not require labeling. The majority of the wide range of methods proposed for sentiment analysis has been supervised machine learning approaches [21]. Furthermore, many attempts have been made to enhance the predictive performance of supervised machine learning classifiers in analyzing the sentiment of tweets in different ways. One such way is using ensemble learning techniques which are a significant subfield of machine learning. Ensemble learning techniques aim to develop classification models with better performance by combining the prediction of different base classifiers into a strong classifier. In producing effective ensemble classifiers, it is crucial to identify base learning classifiers that can perform the classification task and ideally involve classifiers with a variety of structures and outputs. Besides, an appropriate combination schema for base learning classifiers is also critical for the performance of ensemble learning approaches [19]. Additionally, combining the well-performing classifiers can be modeled as an optimization problem; hence, the well-established means of meta-heuristic algorithms can provide optimal solutions. Meta-heuristics based on population encompass particle genetic algorithms, swarm optimization, differential evolution, and ant colony optimization algorithms. Among the many approaches used for sentiment analysis, machine learning-based approaches and meta-heuristic algorithms have been successfully implemented in optimizing ensemble classifier approaches [22, 23]. In recent years, different ensembling approaches have been proposed and applied for sentiment analysis and critically evaluated. The development of GenSent presents a significant advancement in sentiment analysis by integrating genetic optimization with ensemble learning. Our approach addresses several key areas in sentiment classification

and optimization, providing the following major contributions:

- This work aims to explore an effective way to conduct fine-grained sentiment analysis. The GenSent framework advances the capability of sentiment analysis by not only distinguishing broad sentiment categories but also by focusing on a detailed 5-point scale. This allows for a more nuanced analysis of sentiments, including extreme opinions, which enhances the accuracy and granularity of sentiment classification. Additionally, the method includes aspect extraction related to sentiments, which provides deeper insights into the context of opinions.
- We introduce an effective framework that generates optimized classifier ensembles using Genetic Algorithms. The current study involves training a diverse pool of 25 classifiers using six machine learning algorithms (ML): Decision Trees (DT), Random Forest (RF), Naive Bayes (NB), Logistic Regression (LR), Support Vector Machines (SVM), and Stochastic Gradient Descent (SGD). These machine learning models are trained with different parameter settings and feature sets to ensure diversity in the classifier pool. A Genetic Algorithm, selects the most effective classifiers to form an optimal ensemble.
- The proposed method enhanced classifier selection by employing Genetic Algorithms to identify the best-performing subset of classifiers from the pool. This approach ensures that the ensemble benefits from the most capable models, improving overall classification performance and reliability.
- The decision-making process in the ensemble is conducted through a weighted majority voting algorithm. This technique combines predictions from the base classifiers, assigning weights based on their performance to ensure that more reliable classifiers have a greater influence on the final decision.
- The proposed system named GenSent is rigorously tested on major datasets on sentiment analysis. Specifically we employed, the three SemEval 2017 datasets from 4A, Task 4B, and Task 4C, and the Stanford Sentiment Treebank SST-2 and SST-5. The framework is benchmarked against existing methods using the same datasets, demonstrating superior performance and achieving results that surpass previously published results in all cases.

The remaining sections are structured as follows: Section 2 reviews the prior research on sentiment analysis. Section 3 introduces the proposed architecture, including the classifiers and features used; Section 4 describes the datasets and experimental setups, Section 5 reports and discusses the results. Finally, Section 6 concludes the paper with a summary of findings and future work directions.

2. RELATED WORK

In recent years, numerous attempts have been made to improve the performance of supervised and deep-learning classifiers in analyzing sentiments using different approaches. A significant amount of the proposed models focused on choosing appropriate machine learning algorithms as classifiers and features for representing sentiment words. More recent works have employed techniques such as Evolutionary Computing, Rough Sets, Swarm Intelligence, Fuzzy Logic, and Neural Networks. Genetic algorithms (GA) are probabilistic search techniques belonging to the class of evolutionary algorithms. Genetic algorithms are mainly used to choose an optimal solution from the set of feasible solutions. Even though genetic algorithms have been applied to various domains, including signature verification, scheduling, timetable, image processing, robot control, routing, information retrieval, and machine learning [46, 47], they have not been used for sentiment analysis until recently. A pioneering work proposed by Ishaq et al. [48] presented an efficient classification framework for sentiment analysis using CNN and Genetic Algorithm. In this work, Semantic features from movie, automobile, and hotel review datasets have been extracted and transformed into vector space using word2vec. A CNN whose parameters were tuned using a Genetic Algorithm, was used to extract opinions.

Cahya et al. [49] proposed a feature-weighted method to enhance the Complement Naïve Bayes (CNB) classifier based on the Genetic Algorithm to analyze sentiment in tweets using the Twitter airline dataset. This work used term frequency weighting to extract features from preprocessed data to produce a document term matrix which was subsequently input to the Genetic Algorithm to select the optimum combination of feature weights based on the correlation between features and class labels. The Naive Bayes classifier was trained using the training set with feature weights. Iqbal et al. [50] developed a hybrid sentiment analysis framework by combining machine learning classifiers with lexicon-based approaches to classify review datasets from the UCI repository. A new genetic algorithm was proposed to reduce the feature set size by developing a modified fitness function in this framework. SentiWordNet dictionary was used in the fitness function to compute the polarity difference between the feature vector and class label. Fatyanosa et al. [51] employed the Genetic Algorithm as a feature selection process to reduce the features in the sentiment analysis. The experiment used a 5-point scale Twitter dataset related to self-driving cars. In this work, NB was trained and then used to classify testing data. F1-score was used as a fitness function for each population in each generation. The results demonstrate that combining algorithms with genetic algorithms improves the ability of classifiers and recognition of minority classes significantly. Keshavarz et al. [52] introduced a model named Adaptive lexicon learning by genetic algorithm (ALGA) to classify the polarity of sentiments using a genetic algorithm to create

lexicons. A parallel approach was proposed for calculating the fitness of ALGA efficiently on Healthcare Reform (HCR), Obama McCain Debate (OMD), Sanders–Twitter Sentiment Corpus, and SemEval datasets. Saidani et al. [53] presented a weighted genetic algorithm to optimize the process of feature selection in analyzing sentiment in tweets. The authors combined a supervised weighting method with a stochastic search method to generate a feature subset that can select and extract the most efficient features. In recent advancements in sentiment analysis, George and Sumathi [68] introduced a genetic algorithm-based hybrid model for enhancing sentiment analysis performance by combining CNNs with RF classifiers. The authors employ a Genetic Algorithm for optimizing the hybrid model's parameters, combining the robust classification capability of RFs with the feature extraction capabilities of CNNs. This innovative combination shows the effectiveness of GA in refining ensemble methods for enhanced sentiment classification performance. This approach highlights a promising direction to leverage evolutionary algorithms in hybrid machine learning systems.

Jain and Jain [70] proposed a hybrid feature selection model for sentiment analysis. The authors combined CHI Square, Information Gain, and GINI Index to find the feature sets and then used the union Set operation to reduce them. These feature sets were optimized using Genetic Algorithms. Four distinct variants of SVM were employed for sentiment classification, utilizing features optimized through Genetic Algorithm. Huang et al. [69] developed an approach that combines sentiment analysis with Genetic Algorithms to improve stock prediction performance using deep learning models. This method combines sentiment data to inform stock forecasts and employs Genetic Algorithms for optimizing model parameters, improving prediction accuracy. This work shows the effectiveness of using sentiment analysis and Genetic Algorithms to improve financial predictive models. Harnain and Gupta [4] introduced a hybrid evolutionary intelligent model named GA-BERT-SVM for improving sentiment analysis. This method combines the SVM classifier, Genetic Algorithms, and bidirectional encoder representations from transformers (BERT). Through the use of BERT embeddings in conjunction with SVM and GA for initial parameter optimization, the approach overcomes frequent drawbacks including overfitting and local minima. The proposed model demonstrated superior performance in predictive sentiment in tweets.

3. MATERIALS AND METHODOLOGY

3.1. GenSent Framework

The GenSent framework addresses the selection of an optimized classifier ensemble for sentiment analysis of social media posts from a pool of classifier ensembles through the evolution of classifier ensembles as described by [32]. Figure 1 presents the overall GenSent framework. The classifier ensembles are represented as chromosomes, where each bit denotes a classifier's participation in the ensemble. The implementation of the GenSent framework was carried out using Python, a widely used programming language for natural language processing tasks. The following Python libraries were used during development:

- NLTK (Natural Language Toolkit) – for tokenization, stemming, and general text preprocessing.
- Stopwords – to remove common stopwords that do not contribute significantly to sentiment.
- Penn Treebank POS Tagger – to tag parts of speech for more informed feature extraction.
- SentiWordNet/Affine Lexicon – for lexicon-based sentiment scoring.

The framework is comprised of four stages: text preprocessing, feature extraction, base classifier training, and optimal ensemble generation using Genetic Algorithm. Each of these sections is explained in the following sections.

3.2. Data Preprocessing

As shown in Figure 1, text preprocessing is the first stage of the proposed framework. At this step, each sentence in the dataset is preprocessed for further processing in the framework. Four main preprocessing operations, tokenization, normalization, stemming, and stop word removal, are applied to the data, as described below:

- Data is tokenized and converted to lowercase.
- All websites and target mentions are replaced with placeholders.
- Punctuation and numbers are removed.
- More than 2 consecutive letters are reduced to 2.
- Stop words are removed. In this work, a list of stop words was created by excluding sentiment words such as "against", "love", etc. from the English stop word list in NLTK [5].
- Words are stemmed into their root form using Porter Stemming.

Following preprocessing, the datasets with imbalanced distributions among the classes are balanced using oversampling of the minority class.

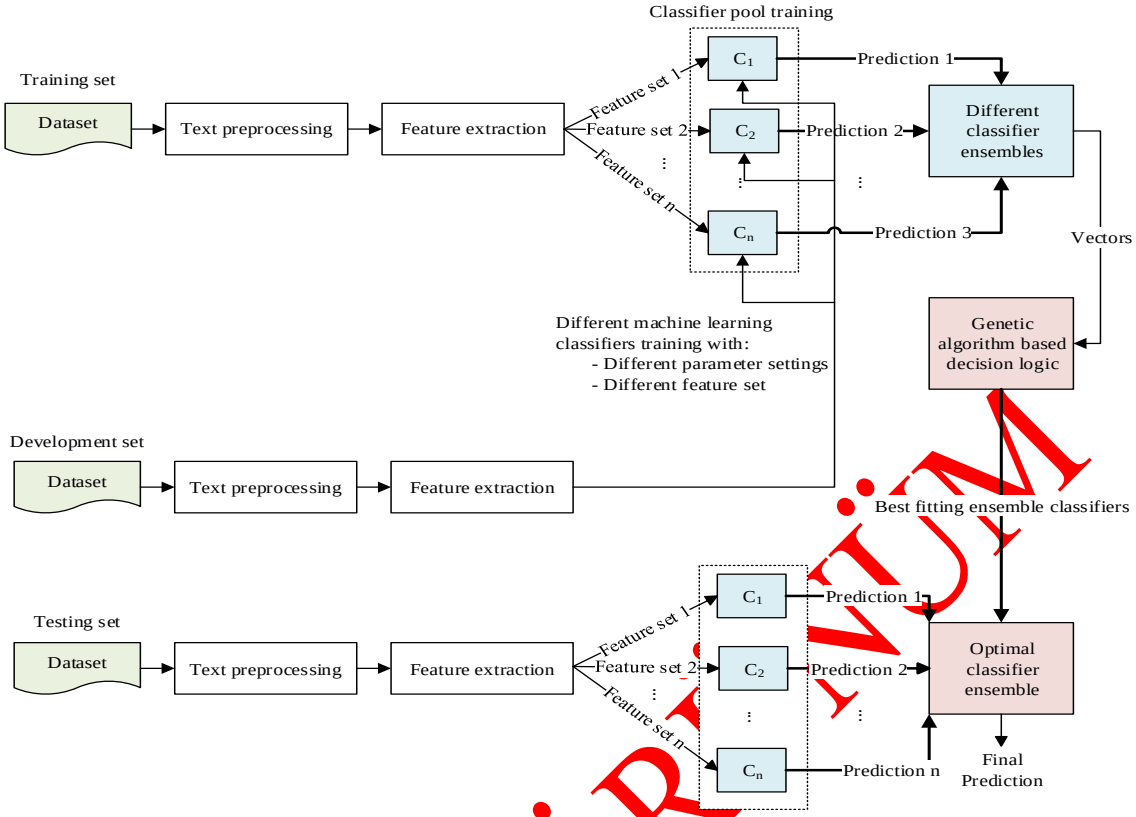

Figure 1. A block diagram of the proposed GenSent framework.

3.3. Feature Extraction

The second stage of the proposed framework of Figure 1 is feature extraction where the features used for training the base classifiers are extracted. The details of the features employed for the current work are presented below:

- Bag of Words (BoW): is a very efficient and flexible method used for extracting features from documents by converting each document to a real-valued vector. To generate BoW vectors, a vocabulary containing all words in a given set of documents is produced. Then, each document is represented as a fixed-length vector where each element shows the occurrence or absence of a vocabulary word in the document.
- Word N-gram: are used to represent the ordering patterns of words within a document. Specifically, bigrams, where n=2, are utilized in the current research to capture sentiments conveyed by consecutive use of two words. In particular, bigrams are expected to contribute to the recognition of negated sentiments such as "not happy", and "not hate".
- Term Frequency-Inverse Document Frequency (TF-IDF): is a natural language process technique that measures how relevant a word is to a document in a collection of documents. The two components of TF-IDF are TF and IDF. Term Frequency (TF) measures the frequency a word appears in a specific document. Inverse Document Frequency (IDF) on the other hand represents how common the word is across all the documents. TF-IDF is calculated by multiplying these two terms.

3.4. Classification Algorithms

This subsection provides brief descriptions of the base classifiers employed in the classifier pool. In this work, we employ the following six classifiers that have been shown to be successful in the sentiment analysis domain by previous research [45,57,58]:

- Support Vector Machine (SVM) has been extensively employed as a machine learning classifiers in sentiment analysis tasks. During the training stage, SVM tries to find the hyperplanes separating data points from different classes by the highest margin thus achieving better generalization over unseen data. During the testing stage, SVM classifies input vectors as positive or negative based on the side of the hyperplane to which they are mapped. The separating hyperplane of SVM is computed using a linear kernel for linearly separable data whereas for non-linearly separable data, it is computed using a nonlinear kernel such as Radial Basis Function (RBF) or Sigmoid that transforms the current features into higher-dimensional feature space. SVM employs the following discriminant function:

$$f(x) = W^N g(x) + b \quad (1)$$

where W^N represents the vectors' weight and g(x) denotes a non-linear mapping between input features to high dimensional features, b presents the term of bias.

- Naive Bayes (NB) is a conditional probability algorithm belonging to the probabilistic-based classifiers family based on Bayes' theorem. It calculates the probability of each class from the provided data, assuming that all features are conditionally independent. This approach enables the prediction of class membership probabilities and helps in filtering out irrelevant information [75]. NB classifier assumes that the features are strongly uncorrelated and computes the probability. In summary, the NB classifier assumes that all features in the feature vector X are mutually independent and computes the probability of class C given the input vector X as:

$$P(C_i|X) = \frac{P(C_i)\,P(X|C_i)}{P(X)} \quad (2)$$

where C_i defines the classes and X denotes the input vector spaces, thus $P(C_i)$ and $P(X)$ are the prior probability of class i and text vector X respectively, accordingly $P(X|C_i)$ is the likelihood which represents the probability of input vectors appearing in a given class i. After computing the conditional probabilities for all classes, the decision of the NB classifier is computed using the following equation:

$$\hat{Y} = argmax_{i\in\{1,\ldots,j\}}\, P(C_j)\prod_{i=1}^{k} P(X_i|C_j) \quad (3)$$

- Logistic Regression (LR) is an efficient and flexible statistical analysis algorithm that belongs to the family of generalized linear models. It is an extension of linear regression methods designed for classification tasks [79]. The LR classifier has been frequently used for sentiment analysis problems, to determine the categorical target variables using one or more predictor variables [44]. Furthermore, LR employs a logistic function to model probabilities, and it effectively estimates the likelihood of an outcome belonging to a particular category. LR classifier is formulated in the following form:

$$P = \frac{1}{1 + e^{-(b_0 + b_1X_1 + \cdots + b_nX_n)}} \quad (4)$$

where P is the predicted probability that the outcome is present, the term b_i(i =0, 1, 2,…, n) represents the regression coefficients, and X_j(j =1, 2,…, n) denotes different independent variables.

- Stochastic Gradient Descent (SGD) is an iterative method that updates model parameters incrementally by computing the loss function's gradient with respect to the parameters. SGD is computationally efficient and has been used successfully for the classification of high-dimensional data. It improves loss functions such as linear SVM and LR classifiers. Although the SGD model has been available for a long time, it recently received a considerable amount of attention due to its performance. SGD formula can be presented as follows:

$$W_{t+1} = W_t - \alpha\frac{\partial L}{\partial W} \quad (5)$$

where, W_{t+1} and W_t are current and old weight respectively, $\partial L/\partial W$ is the current gradient multiplied by some factor α called the learning rate used to update W_{t+1}.

- Decision Tree (DT) is one of the most well-known learners that have been used successfully for various tasks. DT classifier does not require any domain knowledge and has the ability to handle both categorical and numerical text data as well as high dimensional and noise data. DT constructs classification models in the form of a tree structure in which data points are broken down into smaller subsets and gradually an associated DT is incrementally constructed. The dataset is sequentially divided by decision tree algorithms. The features that work effectively during classification are used to identify the initial condition. The end result of this process shows a tree with the decision and leaf nodes, the top of the decision node in a tree is called the root node which corresponds to the best predictor. It also depicts that if a particular sequence of outputs occurs then which decision node has the highest probability to occur and what class label will be assigned for that sequence. The main idea behind DT is the use of the Iterative Dichotomiser 3 (ID3) algorithm which utilizes Information Gain (IG) and Entropy function for constructing a DT. The following formula shows using the concepts of the Entropy function to find the split point and the feature to split on, and mathematically it can be written as:

$$E(\delta) = -\sum_{i}^{n} P(C_i)\,log_2\,(P(C_i)) \quad (6)$$

where i represents the number of features; $P(C_i)$ is the probability of class C_i in a dataset; and δ represents the target class.

- Random Forest (RF) is also known as random decision forest. It is an ensemble learning algorithm that can be used for both regression and classification tasks. RF algorithm constructs a

number of DT models; the combination of those multiple DT models results in a forest of DTs. Therefore it can be termed as the collection of tree-structured classifiers. In the beginning, RF trains DT classifiers where each tree is constructed using a random subset of different vector features. After that, the sequence of vector features and their values generate a route to leaves which represent the decisions. Then the decisions of all trees are fitted into a meta-estimator to make a forest. RF uses a majority voting algorithm to derive the resultant class label from the generated classes through similar subsampled trees generated as the RF outcome. In FR at training time, the decision node values are updated to reduce a cost function that estimates the performance of the trees. In addition, RF decreases variance by training each DT on different samples of the dataset and utilizing random samples of different vector features [42, 43]. Furthermore, the use of more trees in the RF algorithm generally corresponds to better performance and produces effective predictive outcomes [42]. In this work, the RF model uses the impurity measure Gini Index to decide how nodes branch in a DT. The Gini index used for building the DTs in RF can be represented as follows:

$$\text{Gini} = 1 - \sum_{i=1}^{n} (P_i)^2 \qquad (7)$$

where P_i denotes the relative frequency of the class observed in the dataset, n is the number of classes.

3.5 Classifier Pool Generation

The classifier pool employed in this work consists of twenty-five classifiers based on six of the most frequently used machine learning algorithms, namely SVM, NB, LR, SGD, RF, and DT. Ensuring a diverse set of classifiers is crucial in the optimized classifier ensemble method because it is not possible to enhance the predictive performance of classification when combining classifiers with the same behavior [33]. The diversity of classifiers can be achieved using different classification or machine learning algorithms and for a given algorithm by training it with different parameter settings and feature combinations. Consequently, in this framework, each base classifier differs from the others in terms of at least one of the following: Machine learning algorithm, parameter values, and features. For instance, the SVM classifier can be trained using different values of parameter settings for the degree of the polynomial kernel, linear kernel, and radial basis function (RBF). Three of the prevalent features for sentiment analysis, BoW, TF-IDF, and Bigram, are used in different feature combinations to train the base classifiers, as shown in Table 1.

3.6. Ensembling Algorithm

In classification tasks, ensembling combines the predictions of multiple algorithms to create a single, optimized predictive model. In this study, one of the most prevalent ensembling methods, weighted majority voting, is used as the ensembling method to generate a strong meta-classifier that balances out the base classifiers' weaknesses on the datasets in sentiment analysis.

The main goal of the weighted majority voting predictor algorithm is to produce an efficient meta-learning classifier that associates each base classifier with a specific weight representing its confidence. In this context, the weights are considered during the process of collecting the votes by increasing and decreasing the impact of base classifiers' predictions based on their accuracy [41, 80]. The formula of the weighted majority voting algorithm can be written as follows:

$$\hat{y} = argmax_i \sum_{j=1}^{n} W_j X_A(C_j(X) = i) \qquad (8)$$

where, W_j represents the weight of j^{th} base classifiers C_j in an ensemble, $\hat{y}$ denotes the predicted class of the ensemble classifiers, X_A is the characteristic function $C_j(X) = i \in A$, and A is the set of unique class labels.

In this work, the validation dataset is used to evaluate the ability of the trained classifiers to predict the sentiments of tweets. The weighted majority ensembling method combines predictions of the base classifiers in each ensemble by using their classification accuracy on the validation set as their confidence or weight.

3.7. Optimizing Classifier Ensembles

At the ensembling phase, the Genetic Algorithm generates or evolves the optimum classifier ensemble from a large pool of classifiers. Figure 2 shows the flow chart of the GenSent framework. The flowchart details the Genetic Algorithm component of Figure 1.
The details of each step of the GA are described below:

1) Representation of Classifier Ensembles: Each chromosome is a 25-bit string representing a classifier ensemble where a value of '1' at any position means the corresponding classifier, from the pool of classifiers described in Table 1, participates in the ensemble. Figure 3 shows a sample 4-bit chromosome representing an ensemble formed from a pool of 4 base classifiers also shown in the figure. The encoding of the example chromosome represents a classifier ensemble formed from SVM2 and RF. The classifier ensembles represented by the chromosomes make predictions using weighted majority voting where the weight of each participating base classifier is set to its accuracy on the training data. The fitness of a chromosome is calculated by computing the accuracy of the ensemble it represents.

2) Initial Population Creation: At this step, an initial population is created by generating chromosomes as random bit strings. As described above each bit corresponds to a base classifier in the classifier pool and its participation in the decision-making process is determined by the value of the bit.
3) Generation of New Population: Starting with the initial population, genetic operators of selection, crossover, and mutation are applied to each population to produce a new population as the next generation in evolution. This process will continue until a stopping condition is reached or the solution converges to an optimal solution. The genetic operations employed in GenSent are as follows:

- Selection: The algorithm first iterates through the population and finds the chromosome with maximum fitness in the selection process. Then it applies the accept-reject algorithm for selection. In this technique, a random chromosome from the population is selected after which another random number is selected ranging from 0 to maximum fitness. If this second random number is less than the index of the randomly selected object from the population then it will be accepted otherwise rejected. Thus, there will be a high probability for the chromosome with maximum fitness to be selected as a parent for the next generation.
- Crossover: the two selected parents pass through the genetic function of crossover. In the crossover, a new child is produced then a random index in the chromosome is selected as the mid-point. The genes from parent A up to the mid-point and the genes from parent B onward from the mid-point are combined to form the chromosome of the new child.
- Mutation: in mutation, the genes of the chromosome are randomly altered based on the mutation rate. If a randomly generated number is less than the mutation rate, a gene at a particular position is replaced with another random bit.
- Replacement and Elite Count: A new population is produced by using the newly generated population and a subset of the chromosomes with the highest fitness values in the current generation. Elite count determines the number of chromosomes that can be copied from the older generation.

Table 1. The base classifiers and their parameter settings and features used for training.

No.	Classifiers	Parameter Settings	BoW	TFIDF	Bigram
1	SVM1	kernel=set(['linear']), degree=set([3]), gamma=set(['auto'])	X	X	X
2	SVM2	kernel=set(['linear']), degree=set([3]), gamma=set(['auto'])	X		X
3	SVM3	kernel=set(['linear']), degree=set([8]), gamma=set(['auto'])		X	X
4	SVM4	kernel=set(['linear']), degree=set([8]), gamma=set([' scale'])	X	X	X
5	SVM5	kernel=set(['linear']), degree=set([8]), gamma=set([' scale'])	X	X	
6	SVM6	kernel=set(['rbf']), degree=set([3]), gamma=set(['auto'])	X	X	X
7	SVM7	kernel=set(['rbf']), degree=set([8]), gamma=set(['auto'])	X		X
8	SVM8	kernel=set(['rbf']), degree=set([3]), gamma=set(['scale'])		X	X
9	SVM9	kernel=set(['rbf']), degree=set([3]), gamma=set(['scale'])	X	X	
10	NB1	-	X	X	X
11	NB2	-	X	X	
12	LR1	penalty = set(['l2']), random_state=set([0])	X	X	X
13	LR2	penalty = set(['l2']), random_state=set([1])	X		X
14	LR3	penalty = set(['l2']), random_state=set([2])	X	X	
15	LR4	penalty = set(['l2']), ,random_state=set([2])	X	X	X
16	RF1	n_estimators=set([100]), max_depth=set([3)], random_state=set([0])	X	X	X
17	RF2	n_estimators=set([100]), max_depth=set([5]), random_state=set([1])	X		X
18	RF3	n_estimators=set([200]), max_depth=set([3]), random_state=set([0])	X	X	X
19	RF4	n_estimators=set([200]), max_depth=set([5]), random_state=set([1])	X		X
20	SGD1	loss=set(['log']), penalty=set(['l2']), max_iter=set([3])	X	X	X
21	SGD2	loss=set(['log']), penalty=set(['l2']), max_iter=set([5])	X		X
22	SGD3	loss=set(['hinge']), penalty=set(['l2']), max_iter=set([5])		X	X
23	SGD4	loss=set(['hinge']), penalty=set(['l2']), max_iter=set([8])	X	X	
24	DT1	n_estimators=set([100]), max_depth=set([3])	X	X	X
25	DT2	n_estimators=set([200]), max_depth=set([5])	X	X	X

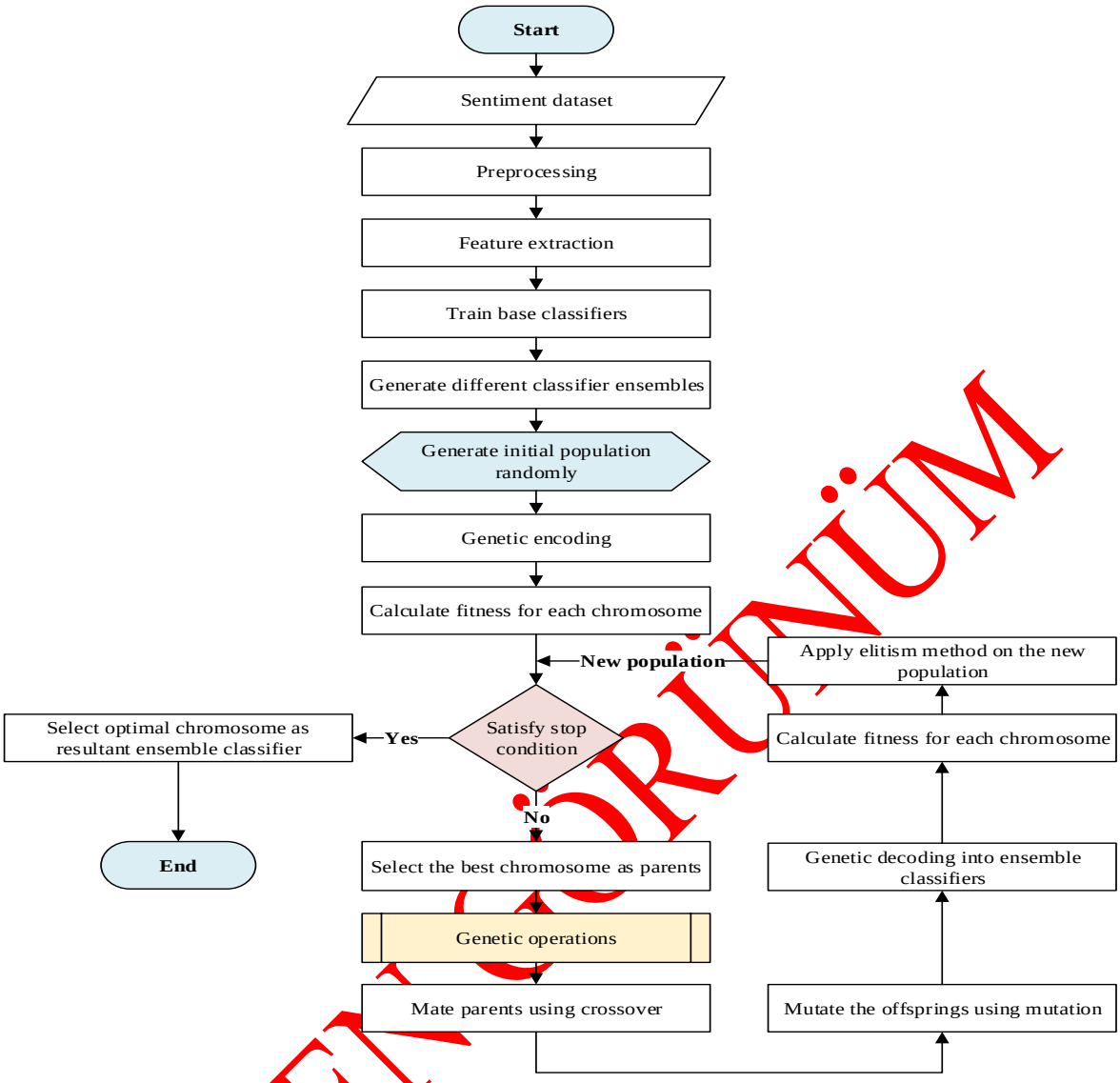

Figure 2. The flowchart of the proposed GenSent scheme

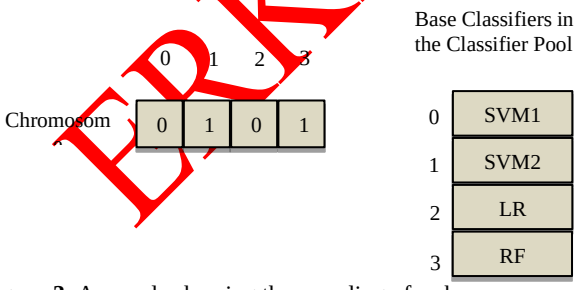

Figure 3. A sample showing the encoding of a chromosome.

4. EXPERIMENTAL RESULTS AND EVALUATION

This section provides an in-depth assessment of results obtained from two experimental sets designed to evaluate the performance and applicability of GenSent. It first describes the development environment, including the datasets, experimental setup, and evaluation process. The results are then analyzed to demonstrate that GenSent outperforms both the highest-performing base classifier and the full ensemble comprising all base classifiers. Additionally, a comparative assessment is provided, highlighting GenSent's performance against existing sentiment analysis methods on the same datasets.

4.1. Evaluation Metrics

In this work, the standard evaluation measures are employed for assessing the performance of the proposed method and compare it to the related work. The total number of correctly predicted positive classes, incorrectly predicted positive classes, correctly predicted negative classes, and incorrectly predicted negative classes are employed to compute these metrics as follows [17]:

$$Accuracy\ (Acc) = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (9)$$

$$Precision\ (Pre) = \frac{True\ Positive}{(True\ Positive + False\ Positive)} \quad (10)$$

$$Recall\ (Rec) = \frac{True\ Positive}{(True\ Positive + False\ Negative)} \quad (11)$$

$$F_1\text{-}Score = 2 \times \frac{Pre \times Rec}{(Pre + Rec)} \quad (12)$$

$$Average\ Recall\ (AveRec) = \frac{1}{3}(Rec^{Positive} + Rec^{Negative} + Rec^{Neutral}) \quad (13)$$

$$F_1^{PositiveNegative}(F_1^{PN}) = \frac{1}{2}(F_1^{Positive} + F_1^{Negative}) \quad (14)$$

$$Macro\ Average\ Mean\ Absolute\ Error\ (MAE^M)\ (h, Te) = \frac{1}{|C|}\sum_{j=1}^{|C|}\frac{1}{|Te_j|}\sum_{X_i \in Te_j}|h(X_i) - y_i| \quad (15)$$

$$Micro\ Average\ Mean\ Absolute\ Error\ (MAE^\mu)\ (h, Te) = \frac{1}{|Te|}\sum_{X_i \in Te}|h(X_i) - y_i| \quad (16)$$

In this context, the actual target associated with X_i is denoted by y_i, and the predicted target is denoted by $h(X_i)$. The set of documents for which the actual class is C_j is represented by Te_j , and the absolute difference $|h(X_i) - y_i|$ correspond to the distance between predicted and actual classes.

4.2. Datasets Used

This section presents the statistics of five publicly available Twitter sentiment datasets used for evaluating GenSent. These datasets are SemEval 2017 Task (4A, 4B, and 4C), and Stanford Sentiment Treebank (SST-2 and SST-5) datasets. Each dataset is split into three partitions: training, development, and test datasets. The number of each class sample in each partition across the datasets used in the experiments is shown in Table 2.

Table 2. Statistics on employed datasets.

Datasets	Type	**SP**	**P**	**Neu**	**N**	**SN**	Total
SemEval-2017 Task 4A	Train	-	3972	5790	1791	-	20632
	Development	-	988	1452	449	-	
	Test	-	2099	3100	991	-	
SemEval-2017 Task 4B	Train	-	4572	-	1336	-	10551
	Development	-	1164	-	313	-	
	Test	-	2476	-	690	-	
SemEval-2017 Task 4C	Train	205	4383	5632	1259	74	20632
	Development	54	1101	1409	308	17	
	Test	123	2346	3040	634	47	
SST-2	Train	-	2759	-	2623	-	9612
	Development	-	679	-	667	-	
	Test	-	1525	-	1359	-	
SST-5	Train	1072	1717	817	1271	1761	11855
	Development	238	436	217	307	462	
	Test	542	958	476	664	917	

[SP: Number of total strongly positive tweets in the data set, P: Number of total positive tweets in the data set, Neu: Number of total neutral tweets in the data set, N: Number of tweets belonging to negative tweets in the datasets, SN: Number of tweets belonging to strongly negative tweets in the datasets.]

4.3. Experimental Procedures

In this section, we provide details about experimental settings for two series of experiments: (i) the ones concerned with the selection of the optimal subset of classifiers from the large pool of classifiers using the Genetic Algorithm; and the selection of the best base classifier in the pool, and (ii) the second assessment of the proposed optimized classifier ensemble with full ensemble classifiers containing all classifiers in the pool. In the experiments, five different datasets are used for evaluating GenSent. The characteristics of these datasets are presented in Table 2. The experimental datasets cover

two binary class problems, one ternary class problem, and two 5-point class problems. The experimental workbench is Jupyter Notebook, a common group of machine learning software written in Python that supports a wide range of workflows in machine learning and data mining tasks. The datasets are split into three disjunctive groups: 50% training set, 20% validation set, and 30% testing set. The use of a single training-validation-test split for evaluation is one of the study's limitations. Although this method is legitimate and widely applied, the randomization of the divide could introduce volatility. By offering performance estimates across several data partitions, k-fold cross-validation may be used in further research to improve the results' robustness and generalizability. Six different classification models trained with using different parameters and features to generate a diverse set of classifiers for the classifier pool. Each chromosome represents an ensemble of classifiers that are combined using weighted majority voting. Each bit in the chromosome corresponds to a classifier and if the bit is 1, the classifier is a member of the ensemble and votes toward the ensemble prediction, otherwise if the bit corresponding to a classifier is 0; it does not contribute to the ensemble decision. The initial population of chromosomes in the Genetic Algorithm is generated randomly. The population size in the simulation experiments is set to 100 chromosomes; each is represented by bit strings of length 25. In this study, the initial population size for the genetic algorithm was set to 100 chromosomes. This value was selected based on a balance between exploration of the solution space and computational efficiency, as commonly suggested in the literature [76, 77]. Previous studies employing GA for optimization in machine learning tasks have used similar population sizes, such as 80 to 150 chromosomes, to maintain adequate diversity without incurring excessive computation [78]. Therefore, 100 chromosomes were deemed an appropriate and practical choice for our ensemble optimization task. Although specific applications of genetic algorithms to ensemble optimization in sentiment analysis are limited, the selected value aligns with settings used in related classification and optimization problems. Thus 100 different ensemble classifiers evolve at the same time. The maximum number of epoch is set to 1000. The accuracy performance given by the weighted majority voting as the combination rule is used to determine the fitness of each chromosome. The Tournament Selection method is employed to select the pair of chromosomes with the highest fitness values from a randomly selected subset of the population. The crossover (one-point) and mutation techniques are applied to the selected chromosomes at a rate of 0.5 and 0.1, respectively. These two processes are carried out to increase the population's diversity, thus, increasing the chances of preventing a convergence to the local optimum. In this framework, the Elitism method is employed where 10% of the best chromosomes from the previous generation are propagated to the new generation, which is not already present in the new generation. This method can rapidly increase the Genetic Algorithm's performance by preserving the fittest chromosomes over the entire population. After a series of evolution and many generations when the termination condition is met, the population's fittest chromosome is considered the best-optimized ensemble classifier solution.

4.4. Result and Discussion

The classification performance of the proposed GenSent framework is presented and compared with relevant high-performing methods using the same dataset.

4.4.1. Experimental results evaluation

To assess the efficiency of the proposed GenSent scheme, we conduct experiments on five widely used datasets in the sentiment analysis domain. Tables 4 – 8 depict the comparison results of the highest-performing base classifier, the full classifier ensemble, and GenSent on each dataset. The weighted voting technique was used for the full ensemble of classifiers. The highest performance on each dataset is given in boldface, while the second-best results appear in italics.

Table 3. Classification Performance Comparison of Base Classifiers, Full Ensemble, and GA-Optimized Ensemble (GenSent) on the SemEval-2017 Task 4A (Ternary Sentiment Classification) Dataset

Classification Scheme	**Classifier(s)**	**Acc%**	**Rec_{macro2}%**	**$F_{1\text{-}macro}$%**
Best Base Classifier	LR3	64.27	58.82	60.25
Ensemble of all classifiers in the pool	Full Ensemble	*65.4*	*59.67*	*61.2*
Proposed GenSent	SVM2, NB1, LR3, DT1	**76.4**	**76.6**	**75.7**

[*Acc: Accuracy, proportion of correctly classified instances out of all instances. Rec_{macro2}: Average Recall among Positive, Neutral, and Negative classes, $F_{1\text{-}macro}$: Average F_1-Score among Positive and Negative classes.*]

Table 3 depicts the comparison results of the highest-performing classifier ensemble generated by the GenSent framework. LR3 classifier is the highest-performing base classifier with Accuracy, Average Recall, and Average F1-Score of 64.27%, 58.82%, and 60.25% respectively. The proposed GenSent framework selected a four-classifier ensemble comprising SVM2, NB1, DT1, and LR3, as the optimal classifier for SemEval-2017, Task 4A dataset. When assessed across multiple performance criteria, the proposed approach demonstrated superior

results, with 76.4% accuracy, 76.6% average recall, and 75.7% average F1-score and exceeded both the strongest base classifier and the full ensemble model. Furthermore, the full ensemble classifier achieves the second-highest performance. This shows that the full ensemble classifier technique is effective in improving the performance of the base classifiers. GenSent improves the second-best performance by 11%, 19.93%, and 14.5% respectively. It is evident that the Genetic Algorithm is efficient in selecting the best-performing classifiers to classify unseen data.

Table 4. Classification Performance Comparison of Base Classifiers, Full Ensemble, and GA-Optimized Ensemble (GenSent) on the SemEval-2017 Task 4B (Binary Sentiment Classification) Dataset

Classification Scheme	**Classifier(s)**	**Acc%**	**Pre%**	**Rec_{macro1}%**	**F_1-Score%**
Best Base Classifier	SVM1	87.2	*91.1*	*79.32*	*91.5*
Ensemble of all classifiers in the pool	Full Ensemble	*88*	88.8	76.63	90.67
Proposed GenSent	SVM1, SVM2, NB1, LR2, RF1 and SGD1	**94.3**	**96.83**	**94.22**	**95.51**

[*Acc: Accuracy, Pre: Precision, proportion of correctly predicted positives out of all predicted positives. Rec_{macro1}: Average Recall among Positive and Negative classes. F_1-Score: harmonic mean of precision and recall*]

As shown in Table 4, the greatest predictive performance for SemEval-2017, Task 4B dataset was obtained by GenSent with the values of 94.3%, 96.83%, 94.22%, and 95.51% respectively. According to the results achieved, SVM1 is a better base classifier in the pool compared to other base classifiers. Notably, the proposed method improves the performance of the second-best algorithm by 6.3%, 5.73%, 14.9%, and 4.01% respectively. As could be seen in Table 4, GenSent selected an ensemble containing six different classifiers, namely SVM1, SVM2, NB1, LR2, RF1, and SGD1 as the optimal classifier on the SemEval-2017, Task 4B dataset. This indicates that the Genetic Algorithm successfully selects the best-performing classifiers in the pool because the SVM1, the best base classifier, yields the highest classification performance.

Table 5. Classification Performance Comparison of Base Classifiers, Full Ensemble, and GA-Optimized Ensemble (GenSent) on the SemEval-2017 Task 4C (Five-Class Sentiment Classification) Dataset

Classification Scheme	**Classifier(s)**	**Acc%**	**F_1-Score%**	**MAE^M**	**MAE^μ**
Best Base Classifier	SGD1	*77.67*	76.04	*0.83*	*1.07*
Ensemble of all classifiers in the pool	Full Ensemble	*80.8*	*80.1*	0.903	1.134
Proposed GenSent	SVM1, SVM2, LR1, SGD1 and DT2	**85.51**	**85.1**	**0.154**	**0.323**

[*Acc: Accuracy, MAE^M: Mean Absolute Error averaged equally across all classes, MAE^μ: Mean Absolute Error calculated over all instances, reflecting class distribution*.]

Table 5 presents Accuracy, F1-Score, Average F1-Score, Macro-average Mean Absolute Error (MAE^M), and Micro-average Mean Absolute Error (MAE^μ) obtained by the best base classifier, the full ensemble, and the ensemble formed by the GenSent framework. These results show that the ensemble formed by GenSent surpasses both the highest-performing base classifier and full ensembleOf particular note is that the proposed method attained the best results when evaluated for accuracy (85.51%) and F1-score (85.1%), and also produced the lowest error values of 0.154 for , MAE^M and 0.323 for MAE^μ. Additionally, the chromosome selected as the optimal ensemble by GenSent contains only five classifiers out of twenty-five classifiers, namely SVM1, SVM2, LR1, SGD1, and DT2. The second-best predictive performance was obtained using the full ensemble classifier method with 80.8% and 80.1% in terms of Accuracy and F1-Score, respectively. The second-best performance in classification was obtained with the single best-performing classifier SGD1 with values of 0.83% and 1.07% in terms of MAE^M and MAE^μ respectively. It should be noted that the full ensemble classifier technique provides higher MAE^M and MAE^μ values when compared to the Accuracy and F1-Score values. This outcome is expected since the optimization process used Accuracy or F1-Score value as the objective function.

Table 6. Classification Performance Comparison of Base Classifiers, Full Ensemble, and GA-Optimized Ensemble (GenSent) on the SST-2 (Binary Sentiment Classification) Dataset

Classification Scheme	Classifier(s)	Acc%	Pre%	Rec%	F_1-Score%
Best Base Classifier	NB1	78.97	78.77	*79.82*	79.29
Ensemble of all classifiers in the pool	Full Ensemble	*79.54*	*82.17*	78.29	*80.18*
Proposed GenSent	SVM2, SVM3, SVM7, SVM8, NB1, LR1, LR3, RF4 and SGD1	**92.6**	**84.2**	**83.3**	**83.74**

[*Acc: Accuracy, Pre: Precision, Rec: Recall: proportion of correctly predicted positives out of all actual positives.*]

Table 6 presents the performance results of the top-performing base classifier, the full ensemble, and the ensemble produced by GenSent on the SST-2 dataset. The highest predictive performance on this dataset was obtained with the proposed GenSent scheme with the values 92.6%, 84.2%, 83.3%, and 83.74, respectively. The full ensemble was ranked as the second-best performance in terms of Accuracy, Precision, and F1-Score, which scored 79.54%, 82.17, and 80.18%, respectively. Meanwhile, the second-best performance in terms of Recall was achieved individually by the best-performing NB1 classifier was 79.82%. GenSent ensemble provides a significant improvement on the second-best performance with respect to Accuracy, Precision, Recall, and F1-Score by 13.06%, 2.03%, 3.48%, and 3.56%, respectively. Additionally, the proposed GenSent classification scheme selected nine classifiers out of twenty-five classifiers, including SVM2, SVM3, SVM7, SVM8, NB1, LR1, LR3, RF4, and SGD1. It can be further observed that classifiers exploit rich feature sets in the classifier set to provide better recognition performances to be selected during the optimization process.

Table 7: Classification Performance Comparison of Base Classifiers, Full Ensemble, and GA-Optimized Ensemble (GenSent) on the SST-5 (Five-Class Sentiment Classification) Dataset

Classification Scheme	Classifier(s)	Acc%	Pre%	Rec%	F_1-Score%
Best Base Classifier	SVM2, SVM3	53.95	*53.62*	53.79	*53.47*
Ensemble of all classifiers in the pool	Full Ensemble	*54.64*	53.17	*54.75*	53.36
Proposed GenSent	SVM2, SVM3, SVM6, RF3 and SGD1	**62.94**	**62.2**	**62.5**	**62.5**

[*Acc: Accuracy, Pre: Precision, Rec: Recall.*]

The performances achieved by the ensemble formed by GenSent, the best base classifier, and the ensemble of all base classifiers for the SST-5 five-point scale dataset are presented in Table 7. Among the 25 classifiers, SVM2 and SVM3 delivered the highest performance; GenSent ensemble with 62.94% Accuracy score, 62.2% Precision, 62.5% Recall, and 62.5% F1-score, respectively surpassed these SVM classifiers as well as the combination of all classifiers. The optimal ensemble formed by GenSent compromises the best-performing base classifiers SVM2 and SVM3, and SVM6, RF3, and SGD1. Therefore it can be deduced that the ensemble has improved the best individual scores.

In summary, the experimental results show a significant improvement compared to all individual classifiers and the full-classifier combination across all sentiment datasets. The results presented here, indicate the significance of the Genetic Algorithm in selecting the well-performing ensemble classifier generally. After evolving the initial population for 100 epochs, the highest-performing ensemble was selected as the optimal GenSent ensemble. Although the total number of classifiers in an ensemble varied depending on the dataset, the numbers ranged between 4 and 9 from a classifier pool of 25, clearly highlighting the need for ensembling using an optimal subset. In all cases GenSent ensemble contained the best base classifier, indicating that GenSent is efficient in selecting the high-performing classifiers.

Furthermore, when the GenSent ensembles produced for the 5 datasets are scrutinized, it can be observed that SVM is the most frequently employed machine learning algorithm with 11 occurrences in 5 ensembles. It is followed by LR which was selected 5 times and SGD which was selected 4 times. Therefore it can be inferred that the SVM classifier significantly performs better than the other base classifiers in the pool. Specifically, the SVM2 with linear kernel and BoW and Bigram features has more contributions. Conversely, the DT classifier which was included in only two of the ensembles has the least contributions in our proposed scheme compared to other base classifiers in the pool.

4.4.2. Comparison of result with related works

To further evaluate the effectiveness of GenSent, we provide a set of comparative results of the ensembles it formed against other relevant works in terms of the performance metrics Accuracy, Precision, Recall, Average Recall, F1-Score, Average F1-Score, Macro-average Mean Absolute Error, and Micro-average Mean

Absolute Error. A comparative analysis between GenSent and existing methods has been conducted. Table 9 presents the recorded performances on the datasets SemEval 2017 Task (4A, 4B, and 4C) and Table 10 shows the results on the SST-2, and SST-5 datasets.

Table 8. Comparison of the proposed GA-Optimized Ensemble Method (GenSent) with existing approaches on optimized method with related work on SemEval-2017 Task 4 A, 4B, and 4C) datasets

Studies	Tasks	Acc%	Rec_{macro}%	F_1-Score%	F_{1macro}%	MAE^M	MAE^μ	References
Cliche, M.	4A	65.8	68.1	-	68.5	-	-	[55]
	4B	89.7	89	88.2	-	-	-	
	4C	-	-	-	-	0.481	0.554	
Baziotis, C. et al	4A	65.1	68.1	-	67.7	-	-	[54]
	4B	86.9	86.1	85.6	-	-	-	
	4C	-	-	-	-	0.555	0.543	
Kolovou, A. et al.	4A	65.9	64.8	-	64.8	-	-	[56]
	4B	86.3	85.6	85.4	-	-	-	
	4C	-	-	-	-	0.623	0.734	
Hama Aziz & Dimililer	4A	-	-	-	-	-	-	[40]
	4B	90.8	88.6	94	-	-	-	
	4C	-	-	-	-	-	-	
Sar-Saifee, B. et al.	4A	81	-	-	-	-	-	[20]
	4B	76	-	74	-	-	-	
	4C	-	-	-	-	-	-	
Das and Pedersen	4A	63	-	-	-	-	-	[30]
	4B	89	-	84	-	-	-	
	4C	-	-	-	-	-	-	
Younesi, R. et al.	4A	80.1	-	-	-	-	-	[31]
	4B	75.3	-	73	-	-	-	
	4C	-	-	-	-	-	-	
Proposed GenSent	4A	**76.4**	**76.6**	-	**75.7**	-	-	-
	4B	**94.3**	**94.22**	**95.51**	-	-	-	
	4C	-	-	-	-	**0.154**	**0.323**	

[*Acc: Accuracy, Rec_{macro}: Average Recall, F_{1macro}: Average F_1-Score among positive and negative classes, MAE^M: Macro-average Mean Absolute Error, MAE^μ: Micro-average Mean Absolute Error*.]

Table 8 presents the performance of GenSent in comparison with results reported in prior studies on the datasets of SemEval-2017 Tasks 4A, 4B, and 4C. As the table shows, GenSent outperforms the previous work. In the SemEval-2017 shared task, the 3 highest performing systems employed CNN, LSTM, and neural networks. Specifically the best-performing system for Task 4A, 4B, and 4C was developed by Cliché [55] based on CNN and LSTM. This comparison shows that GenSent outperforms the best-reported results on all tasks 4A, 4B, and 4C of SemEval-2017 datasets. Specifically, the proposed method surpasses the performance of the highest-ranking system by 10.6%, 8.5%, and 7.2% with respect to Accuracy, Average Recall, and Average F1-Score respectively for SemEval-2017 Task 4A 3-point scale classification. Further, our method surpasses the best existing related method by 4.6%, 5.22%, and 7.31% in terms of Accuracy, Average Recall, and F1-Score respectively for SemEval-2017 Task 4B 2-point scale classification. Similarly, an improvement of 0.32.7 in MAE^M and 0.231 in MAE^μ is observed for the proposed system over the highest performing system on the 5-point scale classification task of SemEval-2017 Task 4C.

Table 9. Accuracy Comparison of the Proposed GA-Optimized Ensemble Method (GenSent) with Existing Methods on SST-2 and SST-5 Datasets

Studies	**SST-2**	**SST-5**	**References**
Tripathi, S. et al.	53.3	-	[39]
Lei, Z. et al.	-	49.7	[67]
Sadr, H. et al.	-	53.42	[65]
Hiyama, Y. et al	73.7	-	[62]
Hassan, A. et al.	-	47.5	[60]
Dong, Y. et al.	-	48.34	[59]
Baktha, K. et al.	81.54	44.61	[63]
Chen, T. et al.	82.3	50.6	[64]
Li, W. et al.	-	50.68	[61]
Lu, Y. et al.	-	47.6	[66]
Giménez, M. et al.	82.45	-	[38]
Sadr, H. et al.	-	51.31	[37]
Kasri, M. et al.	-	48.7	[36]
Xu, Y. et al.	81.8	-	[35]
Park and Ahn	80.9	-	[34]
Hama Aziz and Dimililer	85.2	-	[40]
Nkhata and Gauch	-	60.48	[71]
Wang, J. et al.	89.8	52.2	[72]
Cao, B. et al.	85.28	51.32	[73]
Proposed GenSent	**92.6**	**62.94**	-

In Table 9, the accuracy results of the proposed and representative competitive systems on SST-2 (2-point scale) and SST-5 (5-point scale) datasets are presented. We observe that the proposed system yields better accuracies compared to the existing methods on both datasets. In this comparison, it is shown that the accuracy of GenSent method is 2.8% higher than the best results on the SST-2 dataset and 2.5% better than the existing method on the SST-5 dataset. This improved performance may be attributed to the use of a diverse set of classifiers in the ensemble produced by the genetic algorithm. Inspection of the ensemble formed by the proposed framework shows that the best-performing classifiers in the classifier pool contribute to the classification decision.

5. CONCLUSION

This work introduces a novel and effective genetic algorithm-based classifier ensembling framework named GenSent that may be used for binary, ternary, and fine-grained sentiment analysis tasks. In this framework, given a pool of classifiers, the genetic algorithm chooses an optimal subset of classifiers for the sentiment classification task. The proposed system in this work employed the machine learning algorithms SVM, LR, NB, SGD, RF, and DT to build base classifiers for the classifier pool. To produce a pool of classifiers, each of these machine learning algorithms is trained using various combinations of parameter settings and feature sets. The experiments were conducted on two binary datasets, one ternary dataset, and two fine-grained sentiment datasets. The base classifiers in the optimized ensemble generated by the genetic algorithm are combined using a simple majority voting algorithm. The results indicate that GenSent yielded better performance for the sentiment analysis task compared to the individual performance of the best base classifier, full weighted majority voting ensemble method, and related work on the same datasets. It is shown that the proposed method is very effective and has a high performance for all settings. Although this research has filled a gap in the field of sentiment analysis by providing a framework that can be adapted to any sentiment analysis problem, further work on the individual components such as the base classifiers, and the voting algorithm is planned. Additionally, the possibility of producing an optimized ensemble classifier method using other evolutionary algorithms will be explored in future studies. To further assess and improve the suggested framework, future comparisons with Transformer-based models like BERT and RoBERTa, which have demonstrated promising efficacy in text classification, as well as hybrid integration techniques with deep learning architectures will be taken into consideration [74].

ACKNOWLEDGEMENT

The authors wishes to thank all those who contributed indirectly to this research and provided valuable insights during the study.

DECLARATION OF ETHICAL STANDARDS

The authors of this article declare that the materials and methods they used in their studies do not require ethics committee approval and/or legal-specific permission.

AUTHORS' CONTRIBUTIONS

Roza Hamaaziz: Planning the experiments, dataset selection and download, implementation of classifiers and the proposed system, visualization and analysis of the results, writing of the manuscript.

Nazife Dimililer: Conceptualization and supervision, planning the experiments, analysis of the results, writing of the manuscript.

CONFLICT OF INTEREST

There is no conflict of interest in this study.

ABBRIVATIONS

GA	Genetic Algorithm
ML	Machine Learning
BoW	Bag of Word
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
SVM	Support Vector Machine
NB	Naive Bayes
LR	Logistic Regression
SGD	Stochastic Gradient Descent
DT	Decision Tree
RF	Random Forest

REFERENCES

[1] Alarifi, A., Alsaleh, M., and Al-Salman, A., "Twitter turing test: Identifying social machines", ***Information Sciences***, *372*: 332-346, (2016).

[2] Öztürk, N., and Ayvaz, S., "Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis.", ***Telematics and Informatics***, *35*(1): 136-147, (2018).

[3] Liu, B., "Sentiment analysis and opinion mining.", ***Synthesis lectures on human language technologies***, 5(1): 1-167, (2012).

[4] Kour, H., and Gupta, M. K., "Hybrid evolutionary intelligent network for sentiment analysis using Twitter data during COVID-19 pandemic.", ***Expert Systems***, *41*(3): e13489, (2024).

[5] Bird, S., Klein, E., and Loper, E., "Natural language processing with Python: analyzing text with the natural language toolkit.", ***O'Reilly Media, Inc***., (2009).

[6] Pang, B., Lee, L., and Vaithyanathan, S., "Thumbs up?: sentiment classification using machine learning techniques.", ***In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Association for Computational Linguistics***, 10: 79-86, (2002).

[7] Turney, P. D., "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews.", **In *Proceedings of the 40th annual meeting on association for computational linguistics- Association for Computational Linguistics***, 417-424, (2002).

[8] Stoyanov, V., & Cardie, C., "Topic identification for fine-grained opinion analysis.", **In *Proceedings of the 22nd International Conference on Computational Linguistics***, Coling, 817-824, (2008).

[9] Kouloumpis, E., Wilson, T., and Moore, J., "Twitter sentiment analysis: The good the bad and the omg!.", **In *Fifth International AAAI conference on weblogs and social media***, Edinburgh, (2011).

[10] Villena-Román, J., Lana-Serrano, S., Martínez-Cámara, E., and González-Cristóbal, J. C., "Tass-workshop on sentiment analysis at sepln.", ***Procesamiento del Lenguaje Natural***, 50: 37-44, (2013).

[11] Go, A., Bhayani, R., & Huang, L., "Twitter sentiment classification using distant supervision.", ***CS224N project report, Stanford***, 1:12, (2009).

[12] Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T., "SemEval-2013 Task 2: Sentiment Analysis in Twitter.", **In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)***, 312-320, (2013).

[13] Rosenthal , S., Ritter , A., Nakov , P., and Stoyanov , V., "SemEval-2014 Task 9: Sentiment Analysis in Twitter.", ***In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)***, 73–80, Dublin, Ireland, (2014).

[14] Nakov, P., Rosenthal, S., Kiritchenko, S., Mohammad, S. M., Kozareva, Z., Ritter, A., ... & Zhu, X., "Developing a successful SemEval task in sentiment analysis of Twitter and other social media texts.", ***Language Resources and Evaluation***, 50: 35-65, (2016).

[15] Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., Stoyanov, V., "SemEval-2015 task 10: Sentiment analysis in Twitter.", ***In: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval '15***, 450–462, Denver, Colorado, USA, (2015).

[16] Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., and Stoyanov, V., "SemEval-2016 task 4: Sentiment analysis in Twitter.", *arXiv preprint arXiv:1912.01973*, (2019).

[17] Rosenthal, S., Farra, N., and Nakov, P., "SemEval-2017 task 4: Sentiment analysis in Twitter.", **In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)***, 502-518, (2017).

[18] Gamal, D., Alfonse, M., M El-Horbaty, E. S., & M Salem, A. B., "Analysis of Machine Learning Algorithms for Opinion Mining in Different Domains.", ***Machine Learning and Knowledge Extraction***, 1: 224-234, (2019).

[19] Liu, X. Y., Zhang, K. Q., Fiumara, G., Meo, P. D., and Ficara, A., "Adaptive Evolutionary Computing Ensemble Learning Model for Sentiment Analysis.", ***Applied Sciences***, 14: 6802, (2024).

[20] Sar-Saifee, B., Tanha, J., & Aeini, M., "A Hybrid Deep Learning Network for Sentiment Analysis on SemEval-2017 Dataset.", **In *2023 28th International Computer Conference, Computer Society of Iran (CSICC)***, 1-7, IEEE, (2023).

[21] Arif, F., and Dulhare, U. N., "A Machine Learning Based Approach for Opinion Mining on Social Network Data.", **In *Computer Communication, Networking and Internet Security***, 135-147, Springer, Singapore, (2017).

[22] Gogna, A., and Tayal, A., "Metaheuristics: review and application.", ***Journal of Experimental & Theoretical Artificial Intelligence***, 25: 503-526, (2013).

[23] Dos Santos, E. M., "Evolutionary algorithms applied to classifier ensemble selection.", ***XLIV SBPO/XVI CLAIO***, 419-430, (2012).

[24] Symeonidis, S., Effrosynidis, D., Kordonis, J., & Arampatzis, A., "DUTH at SemEval-2017 Task 4: a voting classification approach for Twitter sentiment analysis.", ***In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)***, 704-708, (2017).

[25] Hasan, A., Moin, S., Karim, A., and Shamshirband, S., "Machine learning-based sentiment analysis for twitter accounts.", ***Mathematical and Computational Applications***, 23: 11, (2018).

[26] Othman, M., Hassan, H., Moawad, R., and Idrees, A. M., "A linguistic approach for opinionated documents summary.", ***Future Computing and Informatics Journal***, 3:152-158, (2018).

[27] Wang, J., and Dong, A., "A comparison of two text representations for sentiment analysis.", ***In 2010 International Conference on Computer Application and System Modeling (ICCASM 2010)***, 11, V11-35, IEEE, (2010).

[28] Kanayama, H., Nasukawa, T., and Watanabe, H., "Deeper sentiment analysis using machine translation technology.", **In** ***COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics***, 494-500, (2004).

[29] Raychev, V., and Nakov, P., "Language-independent sentiment analysis using subjectivity and positional information.", ***arXiv preprint arXiv:1911.12544***, (2019).

[30] Das, R. K., & Pedersen, D. T., "SemEval-2017 Task 4: Sentiment Analysis in Twitter using BERT.", ***arXiv preprint arXiv:2401.07944***, (2024).

[31] Younesi, R. T., Tanha, J., Namvar, S., and Mostafaei, S. H., "A CNN-BiLSTM based deep learning model to sentiment analysis.", **In** ***2024 20th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP)***, 1-6, IEEE, (2024).

[32] Dimililer, N., Varoğlu, E., and Altınçay, H., "Vote-based classifier selection for biomedical NER using genetic algorithms.", ***In Iberian Conference on Pattern Recognition and Image Analysis***, 202-209, Springer, Berlin, Heidelberg, (2007).

[33] Dimililer, N., Varoğlu, E., and Altınçay, H., "Classifier subset selection for biomedical named entity recognition.", ***Applied Intelligence***, 31(3):267-282, (2009).

[34] Park, D., and Ahn, C. W., "Self-Supervised Contextual Data Augmentation for Natural Language Processing.", ***Symmetry***, 11(11): 1393, (2019).

[35] Xu, Y., Li, L., Gao, H., Hei, L., Li, R., and Wang, Y., "Sentiment classification with adversarial learning and attention mechanism.", ***Computational Intelligence***, 37(2): 774-798, (2021).

[36] Kasri, M., Birjali, M., and Beni-Hssane, A., "Word2Sent: A new learning sentiment-embedding model with low dimension for sentence level sentiment classification.", ***Concurrency and Computation: Practice and Experience***, 33(9): e6149, (2021).

[37] Sadr, H., Solimandarabi, M. N., Pedram, M. M., and Teshnehlab, M., "A Novel Deep Learning Method for Textual Sentiment Analysis.", ***arXiv preprint*** arXiv:2102.11651, (2021).

[38] Giménez, M., Palanca, J., and Botti, V., "Semantic-based padding in convolutional neural networks for improving the performance in natural language processing. A case of study in sentiment analysis.", ***Neurocomputing***, 378: 315-323, (2020).

[39] Tripathi, S., Singh, C., Kumar, A., Pandey, C., and Jain, N., "Bidirectional transformer based multi-task learning for natural language understanding.", ***In International Conference on Applications of Natural Language to Information Systems***, 54-65, Springer, Cham, (2019).

[40] Hama Aziz, R. H., and Dimililer, N., "SentiXGboost: enhanced sentiment analysis in social media posts with ensemble XGBoost classifier.", ***Journal of the Chinese Institute of Engineers***, 1-11, (2021).

[41] Kuncheva, L. I., "Combining pattern classifiers: methods and algorithms.", ***John Wiley & Sons***, (2014).

[42] Kirasich, K., Smith, T., and Sadler, B., "Random Forest vs logistic regression: binary classification for heterogeneous datasets.", ***SMU Data Science Review***, 1(3): 9, (2018).

[43] Breiman, L., "Random forests.", ***Machine learning***, 45(1): 5-32, (2001).

[44] Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X., "Applied logistic regression", ***John Wiley & Sons***, 398, (2013).

[45] Al Amrani, Y., Lazaar, M., and El Kadiri, K. E., "Random forest and support vector machine based hybrid approach to sentiment analysis.", ***Procedia Computer Science***, 127: 511-520, (2018).

[46] Kraft, D. H., Petry, F. E., Buckles, B. P., and Sadasivan, T., "The use of genetic programming to build queries for information retrieval.", ***In Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence***, 468-473, IEEE, (1994).

[47] Martin-Bautista, M. J., Larsen, H. L., Nicolaisen, J., and Svendsen, T., "An approach to an adaptive information retrieval agent using genetic algorithms with fuzzy set genes.", **In** ***Proceedings of 6th International Fuzzy Systems Conference***, 3: 1227-1232, IEEE, (1997).

[48] Ishaq, A., Asghar, S., and Gillani, S. A., "Aspect-Based Sentiment Analysis Using a Hybridized Approach Based on CNN and GA.", ***IEEE Access***, 8: 135499-135512, (2020).

[49] Cahya, R. A., Adimanggala, D., and Supianto, A. A., "Deep Feature Weighting Based on Genetic Algorithm and Naïve Bayes for Twitter Sentiment Analysis.", **In** ***2019 International Conference on Sustainable Information Engineering and Technology (SIET)***, 326-331, IEEE, (2019).

[50] Iqbal, F., Hashmi, J. M., Fung, B. C., Batool, R., Khattak, A. M., Aleem, S., and Hung, P. C., "A hybrid framework for sentiment analysis using genetic algorithm based feature reduction.", ***IEEE Access***, 7: 14637-14652, (2019).

[51] Fatyanosa, T. N., Bachtiar, F. A., and Data, M., "Feature Selection using Variable Length Chromosome Genetic Algorithm for Sentiment Analysis.", ***In 2018 International Conference on Sustainable Information Engineering and Technology (SIET)***, 27-32, IEEE, (2018).

[52] Keshavarz, H., Abadeh, M. S., and Almasi, M., "A new lexicon learning algorithm for sentiment analysis of big data.", **In** ***2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)***, 000249-000254, IEEE, (2017).

[53] Saidani, F. R., and Rassoul, I., "A weighted genetic approach for feature selection in sentiment analysis.", ***International Journal of Computational Intelligence and Applications***, 16(02): 1750013, (2017).

[54] Baziotis, C., Pelekis, N., and Doulkeridis, C., "Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis.", **In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)***, 747-754, (2017).

[55] Cliche, M., "Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms.", ***arXiv preprint***, arXiv:1704.06125, (2017).

[56] Kolovou, A., Kokkinos, F., Fergadis, A., Papalampidi, P., Iosif, E., Malandrakis, N., Palogiannidi, E., Papageorgiou, H., Narayanan, S. and Potamianos, A., "Tweester at SemEval-2017 Task 4: Fusion of Semantic-Affective and pairwise classification models for sentiment analysis in Twitter.", **In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)***, 675-682, (2017).

[57] Yadav, N., Kudale, O., Gupta, S., Rao, A., and Shitole, A., "Twitter Sentiment Analysis Using Machine Learning for Product Evaluation.", **In *2020 International Conference on Inventive Computation Technologies (ICICT)***, 181-185, IEEE, (2020).

[58] Moh, M., Gajjala, A., Gangireddy, S. C. R., and Moh, T. S., "On multi-tier sentiment analysis using supervised machine learning.", **In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)***, 1: 341-344, IEEE, (2015).

[59] Dong, Y., Fu, Y., Wang, L., Chen, Y., Dong, Y., and Li, J., "A sentiment analysis method of capsule network based on BiLSTM.", ***IEEE Access***, 8: 37014-37020, (2020).

[60] Hassan, A., and Mahmood, A., "Deep learning approach for sentiment analysis of short texts.", ***In 2017 3rd international conference on control, automation and robotics (ICCAR)***, 705-710, IEEE, (2017).

[61] Li, W., Zhu, L., Shi, Y., Guo, K., and Zheng, Y., "User reviews: Sentiment analysis using lexicon integrated two-channel CNN-LSTM family models.", ***Applied Soft Computing***, 106435, (2020).

[62] Hiyama, Y., and Yanagimoto, H., "Word polarity attention in sentiment analysis.", ***Artificial Life and Robotics***, 23(3): 311-315, (2018).

[63] Baktha, K., and Tripathy, B. K., "Investigation of recurrent neural networks in the field of sentiment analysis.", **In *2017 International Conference on Communication and Signal Processing (ICCSP)***, 2047-2050, IEEE, (2017).

[64] Chen, T., Xu, R., He, Y., and Wang, X., "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN.", ***Expert Systems with Applications***, 72: 221-230, (2017).

[65] Sadr, H., Pedram, M. M., and Teshnehlab, M., "A robust sentiment analysis method based on sequential combination of convolutional and recursive neural networks.", ***Neural Processing Letters***, 50(3): 2745-2761, (2019).

[66] Lu, Y., Rao, Y., Yang, J., and Yin, J., "Incorporating Lexicons into LSTM for sentiment classification.', **In *2018 International joint conference on neural networks (IJCNN)***, 1-7, IEEE, (2018).

[67] Lei, Z., Yang, Y., and Yang, M., SAAN: A sentiment-aware attention network for sentiment analysis. In *The **41st International ACM SIGIR Conference on Research & Development in Information Retrieval*** 1197-1200., (2018).

[68] George, C. S., and Sumathi, B., "Genetic Algorithm Based Hybrid Model of Convolutional Neural Network And Random Forest Classifier For Sentiment Classification.", ***Turkish Journal of Computer and Mathematics Education***, 12(2): 3216-3223, (2021).

[69] Huang, J. Y., Tung, C. L., and Lin, W. Z., "Using social network sentiment analysis and genetic algorithm to improve the stock prediction accuracy of the deep learning-based approach.", ***International Journal of Computational Intelligence Systems***, 16(1): 93, (2023).

[70] Jain, A., and Jain, V., "Sentiment classification using hybrid feature selection and ensemble classifier.", ***Journal of Intelligent & Fuzzy Systems***, 42(2): 659-668, (2022).

[71] Nkhata, G., & Gauch, S. "Fine-tuning BERT with Bidirectional LSTM for Find-Grained Movie Reviews Sentiment Analysis". ***International Journal On Advances in Systems and Measurements***., (2023).

[72] Wang, J., Zhang, Y., Yu, L. C., and Zhang, X., "Contextual sentiment embeddings via bi-directional GRU language model.", ***Knowledge-Based Systems***, 235: 107663, (2022).

[73] Cao, B., Jiang, K., and Fan, J., "SLaNT: A Semi-supervised Label Noise-Tolerant Framework for Text Sentiment Analysis**.", In *Proceedings of the International AAAI Conference on Web and Social Media***, 18: 191-202, (2024).

[74] Aydın, N., Erdem, O. A., and Tekerek, A., "Comparative analysis of traditional machine learning and transformer-based deep learning models for text classification.", ***Journal of Polytechnic (Politeknik Dergisi)***, 28(2): 445-452, (2025).

[75] Tunç, Ü., Atalar, E., Gargı, M. S., and Aydın, Z. E., "Classification of fake, bot, and real accounts on instagram using machine learning.", ***Politeknik Dergisi***, 27(2): 479-488, (2022).

[76] [76] Hassanat, A., Almohammadi, K., Alkafaween, E. A., Abunawas, E., Hammouri, A., and Prasath, V. S., "Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach.", ***Information***, 10(12): 390, (2019).

[77] Azedou, A., Amine, A., Kisekka, I., and Lahssini, S., "Genetic algorithm optimization of ensemble learning approach for improved land cover and land use mapping: Application to Talassemtane National Park.", ***Ecological Indicators***, 177: 113776, (2025).

[78] Huang, J. Y., Tung, C. L., and Lin, W. Z., "Using social network sentiment analysis and genetic algorithm to improve the stock prediction accuracy of the deep learning-based approach.", ***International Journal of Computational Intelligence Systems***, 16(1): 93, (2023).

[79] Demirel, U., and Çam, H., "Investigation of fluctuations incryptocurrency transactions with sentiment analysis." ***Politeknik Dergisi***, 28(3): 773–784, (2025).

[80] Aziz, R. H. H., and Dimililer, N., "Twitter sentiment analysis using an ensemble weighted majority vote classifier.", ***In 2020 International Conference on Advanced Science and Engineering (ICOASE)***, 103-109, IEEE, (2020).