Iranian Journal of Science and Technology, Transactions of Electrical Engineering
https://doi.org/10.1007/s40998-024-00718-w

RESEARCH PAPER

An Enhanced Ensemble Learning Method for Sentiment Analysis based on Q-learning

Mohammad Savargiv[1] · Behrooz Masoumi[1] · Mohammad Reza Keyvanpour[2]

Received: 28 September 2021 / Accepted: 4 March 2024

Abstract
Ensemble learning is a powerful technique for combining multiple classifiers to achieve improved performance. However, the challenge of applying ensemble learning to dynamic and diverse data, such as text in sentiment analysis, has limited its effectiveness. In this paper, we propose a novel reinforcement learning-based method for integrating base learners in sentiment analysis. Our method modifies the influence of base learners on the ensemble output based on the problem space, without requiring prior knowledge of the input domain. This approach effectively manages the dynamic behavior of data to achieve greater efficiency and accuracy. Unlike similar methods, our approach eliminates the need for basic knowledge about the input domain. Our experimental results demonstrate the robust performance of the proposed method compared to traditional methods of base learner integration. The significant improvement in various evaluation criteria highlights the effectiveness of our method in handling diverse data behavior. Overall, our work contributes a novel reinforcement learning-based approach to improve the effectiveness of ensemble learning in sentiment analysis.

Keyword Ensemble learning. Reinforcement learning. Sentiment analysis. Base learners integration

1 Introduction

Ensemble learning is a machine learning method whose main idea is to create a cumulative classifier with a higher resolution than individual base learners. There are various ways to integrate base learners in the literature, including majority voting, simple voting, weighted voting, etc. (Werbin-Ofir et al. 2019). Conventional methods for incorporating the results of base learners are desirable when the data behavior is static. However, when data behavior becomes dynamic, diverse, and even antagonistic due to different domains, traditional methods lose their effectiveness.

Text data exhibits different behaviors under different conditions in various domains. The term "domain" refers to the field of application. For example, in the electronics domain, the token "small" has a positive polarity, while in the restaurant domain, it has a negative polarity. The term "dynamic behaviour" also refers to different categories in which a property can fit into different conditions. The presence of an intensifying or attenuating token has a significant impact on determining the level of polarity of a sentence or a document.

To mitigate the impact of these challenges on ensemble learning-based approaches, this paper proposes a reinforcement learning-based method. The purpose of this paper is to present a method for integrating base learners with adaptability to problem conditions and domain-independent capability. In the proposed method, the Q-learning algorithm is applied. The reason for applying this method is that the Q-learning algorithm does not require basic knowledge of the problem space and guarantees global optima of the solution space.

✉ Behrooz Masoumi
Masoumi@Qiau.ac.ir

Mohammad Savargiv
Savargiv@Qiau.ac.ir

Mohammad Reza Keyvanpour
Keyvanpour@alzahra.ac.ir

1 Department of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

2 Department of Computer Engineering, Alzahra University, Tehran, Iran

Published online: 12 April 2024

Our contribution is twofold. First, we apply the feature of not requiring basic knowledge of the problem space to make the proposed method operate independently of the domain. Second, we eliminate the restriction of data-driven by using Q-learning. Since the Q-learning algorithm chooses the best action from the set of selectable actions, it can guarantee finding global optima in the problem space. Therefore, integrating base learners with the Q-learning algorithm ensures achieving the highest quality of performance after episode convergence.

Given that the proposed method integrates base learners with the concept of reinforcement learning, the dynamic nature of the problem space does not affect the ensemble output. At each step of the learning process in reinforcement learning, changes for enhancing and optimizing the classification algorithms are made based on the problem conditions.

Our contributions are evaluated in the evaluation section. The rest of the paper is organized as follows. In Sect. 2, we present related works. Q-learning is introduced in Sect. 3, and the proposed method is explained in Sect. 4. Section 5 presents the evaluation, and the conclusion and future work are presented in Sect. 6.

2 Related Works

In this section, previous work has been examined in two sections. The first subsection contains an overview of ensemble learning and the different methods of integrating basic classifiers. The second subsection discusses various sentiment analysis techniques.

2.1 Ensemble Learning

Ensemble learning methods aim to improve efficiency and increase the accuracy of the identification process by aggregating base classifiers. The three main categories of ensemble learning are Bagging, Boosting, and Random Subspace. Table 1 summarizes the pros and cons of different ensemble learning approaches.

Ensemble implementation styles include homogeneous and heterogeneous base learners. In the homogeneous style, all base learners follow the same classification algorithm and have the same resolution. In contrast, heterogeneous base learners can be selected from different classification algorithms, and their performance may differ, requiring weighting based on their performance.

The main advantage of ensemble learning is the ability to integrate the strengths of multiple base learners, allowing for greater flexibility in creating ensemble learning models. However, assigning weights to base learners during processing is a significant challenge. Simple methods such as assigning constant weights or using majority voting may not be effective in all situations. More sophisticated methods that assign dynamic weights based on the performance of base learners during processing can be used. Table 2 summarizes the pros and cons of different methods for assigning weights to base learners.

Overall, ensemble learning is a powerful tool for improving the accuracy and efficiency of machine learning models. However, the choice of ensemble approach, base learners, and weight assignment methods should be carefully considered based on the specific problem and dataset at hand.

Based on the subject literature presented in this section, a category is shown in Fig. 1. The presented category is based on the reviewed references, and the authors do not claim that all relevant references have been reviewed. However, we have tried to point out the latest and most relevant references in the literature.

2.2 Sentiment Analysis

The text data type is one of the most used data types of social networks. A huge amount of this type of data is transmitted daily among social media users. The concepts conveyed using the text are important. These concepts include sentiment, emotion, affect, and opinion. The terminologies in this field are often mistakenly used interchangeably. The sentiment is formed when a subject is fully understood, and the thinking about that subject

Table 1 Pros and cons of different ensemble learning approaches

Ensemble approach	Pros	Cons
Bagging (Ju et al. 2023)	Reduces variance and overfitting, good for small datasets	Limited diversity among base learners
Boosting (Menor-Flores and Vega-Rodríguez 2023)	Good Performance on imbalanced datasets, can handle noisy data	Prone to overfitting
Random subspace (Sagi and Rokach 2018; Deegalla et al. 2022)	Can handle high-dimensional data, good for feature selection	Limited diversity among base learners

Table 2 Pros and cons of different methods for assigning weights to base learners

Weight assignment method	Pros	Cons
Static weighting (Cherubin 2019)	Simple and efficient	Reinforces both the strengths and weaknesses of classifiers
Dynamic weighting based on learning automata (Savargiv et al. 2022)	Can handle heterogeneous base learners, improves accuracy and diversity of ensemble	Complex algorithm, may require significant computational resources
Cost-sensitive feature selection using genetic algorithm (Y.Aram et al. 2023)	Effective for feature selection in SVM, improves accuracy	May require significant computational resources
Distributed ensemble learning using Fog computing (Kumar et al. 2021)	Improves prediction performance, reduces overhead	May require specialized hardware and software
RF-CatBoost ensemble (Banik and Biswas 2023)	Effective for complex data classification, improves accuracy	May require significant computational resources
Reinforcement learning (Shaw et al. 2021)	Reduces energy costs, improves efficiency	May require significant computational resources
Ensemble classifiers for dealing with imbalanced data (Priya and Uthra 2021)	Effective for handling concept drift and imbalanced data, improves accuracy	May require significant computational resources
HMM-based recognition system (Wang and Yan 2020)	Effective for recognizing cross-view gait, improves accuracy	Limited diversity among base learners
Ensemble learning for recognizing skeleton-based 3D action (Liang et al. 2019)	Effective for recognizing 3D action, improves accuracy	Limited diversity among base learners
Ensemble feature selection (Abasabadi et al. 2021)	Effective for feature selection, improves accuracy	May require significant computational resources
Distributed ensemble learning (Hajewski and Oliveira 2019)	Reduces overhead, improves efficiency	May require specialized hardware and software
Ensemble-based model for analyzing and classifying student data (Vidhya and Vadivu 2021)	Effective for educational objectives, improves accuracy	Limited diversity among base learners
Weighing base learners based on dataset in image processing (Jiang et al. 2019)	Effective for disease recognition and classification, improves accuracy	Limited diversity among base learners

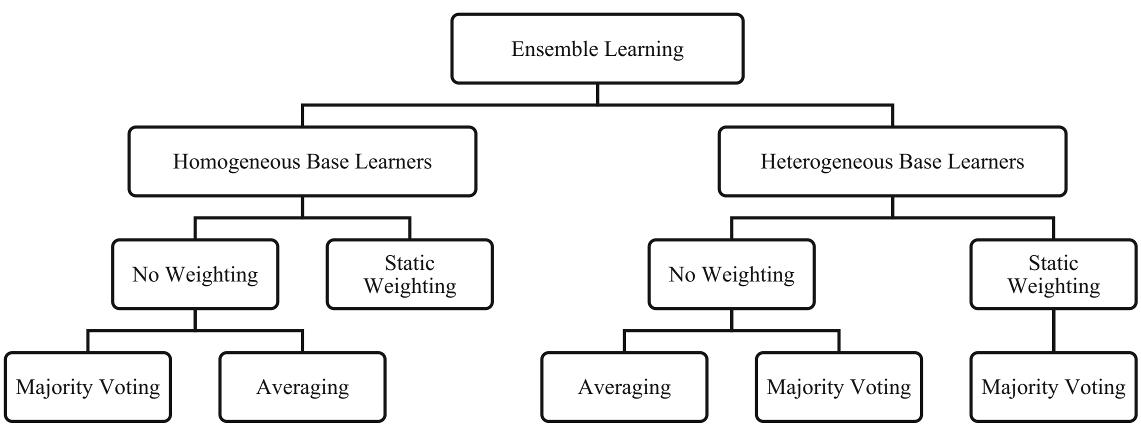


Fig. 1 The ensemble learning categorization diagram in terms of integration method (Savargiv et al. 2022)

evolves (Chaturvedi et al. 2018). Emotional tendencies are formed by the sentiment when a person is involved in a subject or encountering a related subject. The expression of sentiment corresponds to a positive or negative statement. The sentiment usually revolves around a subject, but the emotion is not limited to one subject. Compared to the opinion, the opinion is the product of personal interpretations of information about a subject, and it is not influenced by social experiences. At the same time, the sentiment is usually influenced by society and expressed through emotion (Munezero et al. 2014). According to the above definition, sentiment analysis is a method for classifying text polarity into three levels of a document, sentence, or aspect (Coletta et al. 2014). In examining a document or sentence, it is assumed that only one sentiment is expressed (Chiong et al. 2018). The terminologies used in sentiment analysis can sometimes be confusing, as they are often used interchangeably. The hierarchical structure of the mentioned terminologies is shown in Fig. 2. However, there is a hierarchical structure to these terminologies, which can be helpful in understanding their relationships and nuances. At the top level of the hierarchy is the term "sentiment," which refers to the overall positive or negative tone of a piece of text. Sentiment analysis algorithms typically assign a numerical score to a text document based on its sentiment, with positive scores indicating a positive sentiment and negative scores indicating a negative sentiment.

There are several related terms that are often used interchangeably but have slightly different meanings. One of these terms is "affect," which refers to the emotional state conveyed by a piece of text. Affect can be positive (e.g., happiness, excitement) or negative (e.g., sadness, anger), and sentiment analysis algorithms often use affect as a proxy for sentiment.

Another related term is "feeling," which refers to the subjective experience of an emotion. Feelings are often described using adjectives, such as happy, sad, angry, or anxious. Sentiment analysis algorithms may attempt to infer feelings from text data by analyzing the language used to describe events or situations.

Emotion is another term that is sometimes used interchangeably with affect and feeling, but it has a slightly different connotation. Emotion refers to a complex psychological and physiological state that is typically triggered by a specific event or situation. Emotions can be positive (e.g., joy, love) or negative (e.g., fear, disgust), and they often involve a range of physiological responses, such as changes in heart rate, breathing, and facial expression.

Finally, there is the term "opinion," which refers to a person's subjective evaluation or judgment of a particular topic or issue. Opinions can be positive or negative, and they can be expressed in a variety of ways, such as through explicit statements or through the use of language that implies a particular stance or attitude.

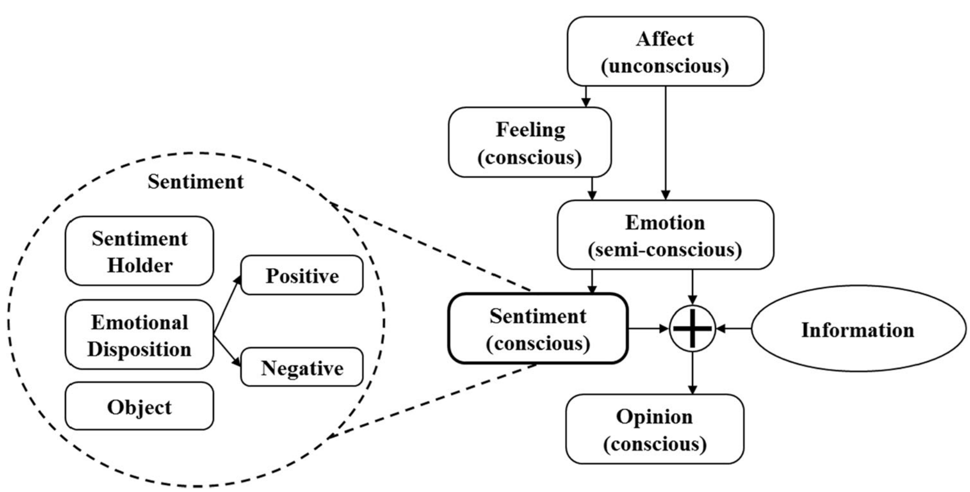
In summary, the hierarchy of sentiment analysis terminologies includes sentiment at the top level, followed by affect, feeling, emotion, and opinion. While these terms are often used interchangeably, understanding their subtle differences can be helpful in developing more nuanced sentiment analysis algorithms that can accurately capture the full range of human emotions and attitudes expressed in text data.

The main purpose of sentiment analysis is to analyze textual comments and determine their polarity (Zhou and Song 2015). Sentiment analysis is a data-driven process that involves various techniques such as natural language processing (NLP) and machine learning models. It has a wide range of applications, including movie box office performance prediction (Sisodia et al. 2020), stock market performance prediction (Li et al. 2020), election forecasting (Sharma et al. 2020), multimodal social media (Huddar et al. 2020), news and blogs (Kaur et al. 2019), recommender systems (Kaya 2020), business intelligence (Sánchez-Núñez et al. 2020), and customer reviews (Basha and Rajput 2019).

Feature extraction is a critical phase in sentiment analysis (Madasu and Elango 2020) because of the challenge of handling ambiguous situations, such as sarcasm. Dictionary-based sentiment analysis techniques typically misinterpret sarcastic text that admires a subject as having negative polarity. Multimodal analysis methods, which use emotional analysis of other types of data such as audio and video in addition to text, are more capable of accurately analyzing such cases (Hernández-Fernández et al. 2019). However, one of the challenges of multimodal analysis methods is how to combine different parts of this method as well as the weighting method for each classifier (Morente-Molinera et al. 2019).

Aspect-based sentiment analysis techniques also require data that is fully organized in specific domains (Ma et al. 2019). If we come across a word with an undefined domain, the method will act as a classical system, losing its

Fig. 2 The hierarchical structure of the sentiment analysis terminologies



aspect-based classification. The challenge of this approach is data dependency.

Similarly, unigram and n-gram analysis methods also require domain-specific data for accurate performance (Ahmed et al. 2020). In both cases, the results of the analysis may be altered if the polarity of a particular word in a particular domain is incorrectly assumed, leading to inaccurate evaluation. Therefore, the accuracy of these methods is highly dependent on the domain of data used in the processing (Dang et al. 2020).

In hierarchical methods of sentiment analysis, determining the hierarchical structure and how to apply it are fundamental challenges (Gargiulo et al. 2019). The access level of each node at different levels of the structure and managing the communication between different levels and nodes are also important (Kumar et al. 2018). In ensemble methods, assigning equal weight to all classifiers in the ensemble is a simple idea (Da'u et al., 2020). However, this approach reduces the strengths of strong classifiers to the level of weak classifiers and increases the weaknesses of weak classifiers to the level of strong classifiers. Therefore, a method to vary the weight of the impacts of the base classifiers under different problem conditions is needed.

As can be seen, traditional methods of sentiment analysis cannot process data with dynamic behavior and all of them require predefined domains. Hence, there is a need for reinforcement learning methods that make decisions based on the problem space. The characteristics of the different methods of sentiment analysis are shown in Table 3.

3 Q-Learning

Q-learning is a reinforcement learning technique that, by learning $<$ state/action $>$ function, follows a specific policy for performing different actions in different states (Chen et al. 2020) and (Sutton and Barto 2018). One of the strengths of this approach is the ability to learn without having a specific model of the environment. In the Q-learning method, the possible actions for the learning agent are identified at each step, and the appropriate states for each of them are determined. Then the reward is given for each action in each state. Based on Eq. (1), a Q value is assigned to each $<$ state/action $>$ pair.

$$Q_{t+1}(s,a) = (1-\alpha)Q_t(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)] \tag{1}$$

In the learning phase, the learning agent fills the Q-table, and in the action phase, it selects its selectable actions from the Q-table in such a way that it has the highest Q-value in each transition from one state to another. The parameter "α" is the learning rate $(0<\alpha\leq 1)$ and $(1-\alpha)Q_t(s,a)$ is the current weighted value. The learning rate determines how much new information is preferred over old information. The "r" is the reward parameter and the discount parameter "γ" determines the importance of future rewards. Zero makes the learning factor opportunistic or shortsighted and only reflects current rewards, while one encourages the learner agent to try for a longer period to reward. $\gamma \max_{a'} Q(s',a')$ is the maximum amount of reward that the next state can receive.

Algorithm 1 illustrates the pseudo-code of the Q-learning algorithm.

Table 3 The characteristics of different methods of sentiment analysis

Method	Characteristics
Dictionary-based method	• Traditional classification method: Domain dependency • Aspect-based sentiment analysis: Requires fully organized data in specific domains (Wu et al. 2021), (Pham and Le 2018), (Song et al. 2021) • Unigram and N-gram sentiment analysis: dependency on the predefined domain of data • Hierarchical sentiment analysis: Hierarchical structure determination and managing communication between different levels and nodes
Reinforcement learning-based method	• No domain dependency (Beigi and Moattar 2021) • No need for basic knowledge about data • Ability to manage data with dynamic behavior

Algorithm 1 The pseudo-code of the Q-learning algorithm

Algorithm Q-Learning

```
Initialize Q_t(s, a) // arbitrarily
Repeat (for each episode)
    Initialize S randomly
    Repeat (for each step)
        Select an action a_i
        Execute the action a
        Observe r (s, a), S′
        Update Q-Table according to Eq. (1)
        s ← s′
    Until S is the terminate state
Until some stopping criteria are reached
```

The general form of Q-learning and a simple schematic of the Q-table is shown in Fig. 3. In the simplest form, the Q-table contains a 2D matrix including state and action respectively. The content of the cells in this table is initially empty, but over time, values are assigned to these cells that are equal to the amount of reward that the learning agent can get in a certain state based on choosing a certain action.

Initialization to Q-table cells can also be done. The learning process (i.e. the process of changing the contents of the Q-table cells) continues until no noticeable change is observed in the cells in successive iterations. To complete the learning process, a predefined number of repetitions can be considered.

In another way, the Q-leaning algorithm can be used stateless, the difference between this mode and the general mode is shown in Eq. (2). (Wilhelmi et al. 2017), (Barrachina-Muñoz et al. 2021)

$$\hat{Q}(a_k) = \hat{Q}(a_k) + \alpha.(r_{ak} + \gamma.\max \hat{Q} - \hat{Q}(a_k)) \quad (2)$$

where, a_k is the set of all possible actions.

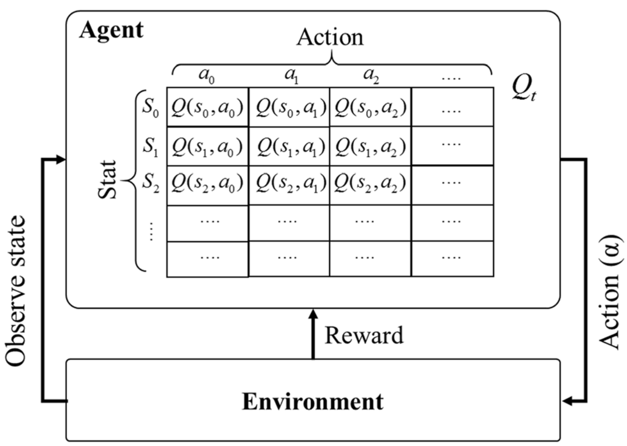
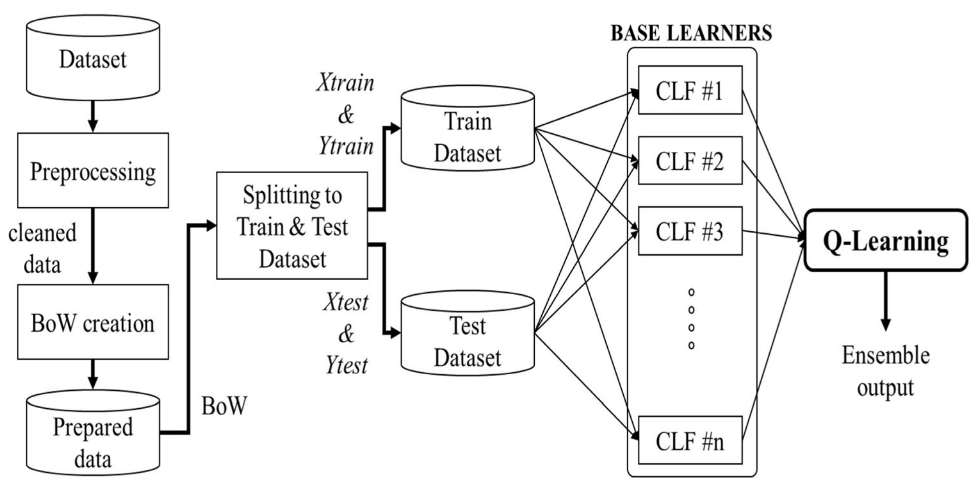


Fig. 3 The general form of Q-learning along with a simple schematic of the Q-table

4 Proposed Method

In this section, the structure of the proposed method is presented along with details. In the proposed method for integrating base learners into the ensemble learning approach, the concept of reinforcement learning and the Q-learning tool is applied. The core idea of reinforcement learning is based on feedback from the environment, so this type of learning is one of the best options for dealing with dynamic issues. At each step, the reinforcement learning algorithm updated its actions based on the problem conditions and adapted to them. The main idea of this article is based on reinforcement learning. Based on the proposed idea, the applicability can be adapted to the problem conditions. In the proposed idea, Q-learning is applied.

Fig. 4 Block diagram of the proposed method



Q-learning does not require basic knowledge of the problem space. In this paper, this feature is the reason for applying this type of reinforcement learning to integrate base learners.

The general procedure of the proposed method is as follows. In each episode of the executable steps, the Q-learning algorithm tries to maximize the sum of achievable future rewards. The set of selectable states of the Q-learning algorithm at each step corresponded to choosing one base learner. A state is chosen if it can provide the possible reward from all different selectable states. The execution process will continue until the maximum reward is obtained or the default number of episodes has been executed. The block diagram of the proposed method is shown in Fig. 4. The details of the proposed method are described below.

4.1 Preprocessing

The preprocessing step in the block diagram is generic, and it will vary according to the type of data. However, as noted, the text data type is one of the types of data that exhibit dynamic behavior under different conditions (i.e., different domains). Therefore, in this paper, this type of data is selected for evaluating the proposed method. Depending on the type of data, appropriate preprocessing is considered. In the preprocessing step, some important preprocessing on the text has been performed to prepare the text for the main processing. The details of the preprocessing step are shown in Fig. 5.

Expressive Lengthening. This preprocessing is also known as word lengthened or word stretching. This preprocessing refers to replacing the original word with words written in the repetition of one of the letters drawn. Written in a stretched form better expresses the emotional state in the text, but they are misspelled.

Emoticons Handling. This preprocessing refers to replacing the emoticons in the text with their corresponding meanings.

HTML Markups Removal. This preprocessing refers to the removal of HTML markups that do not evoke any sentiment.

Slangs Handling. In this preprocessing, slang words are replaced with the original word.

Punctuation Handling. All punctuations and numbers except apostrophes are removed in this preprocessing.

Stopwords Removal. This preprocessing refers to the removal of all words that have no emotional meaning.

Stemming. This preprocessing refers to removing all prefixes and suffixes and replacing the word with its root form.

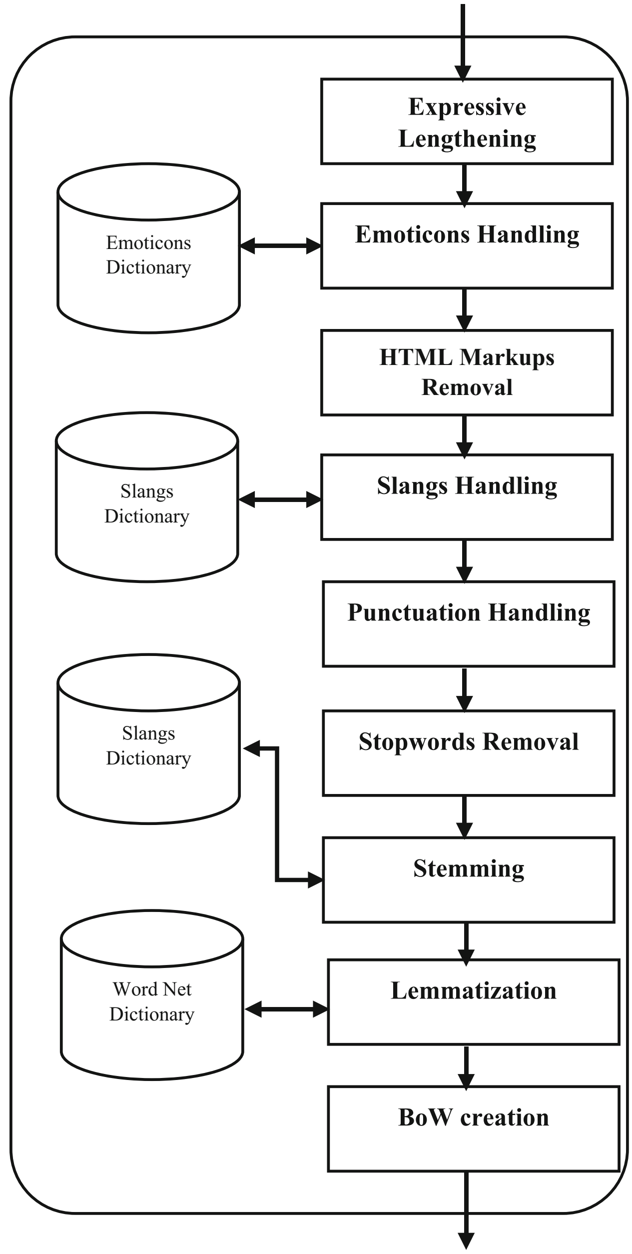


Fig. 5 Text preprocessing and preparing steps

Lemmatization. This preprocessing is similar to stemming, in which the dictionary form and its synonyms are replaced.

BoW creation. This preprocessing is the last preprocessing in text preparation, in which the corresponding numerical matrix of words is formed.

The preprocessing process is implemented using the Python library. Porter's algorithm is used for the stemming step, which has high efficiency, Also, the Wordnet Lemmatizer approach is used for lemmatization, which supports more than 200 languages and is one of the oldest and most common techniques. Table 4 shows the goals and examples

Table 4 Examples of preprocessing considered for text preparation

Preprocessing	Description	Example	
		Input	Output
Tokenization	Break a sentence into words, phrases, symbols, or other meaningful tokens	“This is my new car”	“This”, “is”, “my”, “new”, “car”
Expressive lengthening	Replaced word with the original form if written in the repetition of one of the letters	“Haaaaappy” “preeeeetty”	“Happy”, “pretty’”
Emoticons handling	Replace emoticons with their meanings	:-), 8–0,:-*	“basic smiley”, “oh my god”, “kiss”
HTML Markups removal	Remove HTML markups	< p >, < / p >, < br >, < /br >	will remove these markups
Slangs handling	Replace slang with their original words	“wassup”, “gr8”, “2mrw”	“what’s up?”, “great”, “tomorrow”
Punctuation handling	Punctuations and numbers are removed except apostrophes	“ n’t “, “ ‘s”, “ ‘re”	“not”, “is”, “are”
Stopwords removal	Remove Stopwords to simplified text	“It was the best movie I have ever seen.”	“best movie ever seen”
Stemming	Remove various prefixes and suffixes, to reduce the number of words	“‘user”, “users”, “used”, “using”	“use”
Lemmatization	Return the base or dictionary form of a word, which is known as the lemma	“looks”, “was”	“look”, “be”

Table 5 An example of BoW creation

Text: [This, is, my, university, one, of, the, best, in, world, I, am, studying, computer, engineering, at]		
review #	Text	Bow
1	This is my university	{1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
2	My University is one of the best one in the world!	{0, 1, 1, 1, 2, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0}
3	I am studying computer engineering at university	{0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1}

before and after each preprocessing, and Table 5 shows examples of the BoW creation process.

4.2 Base Learners

After dividing the dataset into two categories of training data and test data in a proportion of 80% and 20%, the dataset is prepared for evaluation. According to the presented brief taxonomy on related work, there are two types of ensembles: heterogeneous base learners and homogeneous base learners. In the proposed method, there is no limitation on the number and type of considered classifiers as base learners, nor is there a limitation on the necessity of homogeneity or heterogeneity of base learners. The proposed method chooses the most complex form to form an ensemble, i.e., heterogeneous base learners. In this case, the different capabilities of different base learners vary, and the dynamic behavior of textual data is more pronounced under these conditions.

4.3 Q-Learning Component

The Q-learning component forms the core of the proposed method. In this component, the integration of base learners is performed using the reinforcement learning idea, and the learning algorithm updates its actions by increasing the amount of reward it can receive. The set of choices that the Q-learning algorithm can make at each step (i.e., the set of actions performed by the Q-learning algorithm) corresponds to choosing one of the base learners that provides the most achievable reward. Initially, the probability of choosing all actions in the Q-table is considered equal, meaning that the learning algorithm does not require basic knowledge of the environment. Therefore, the limitation of data-driven and dynamic behavior of data across different domains does not affect the result of the ensemble classification algorithm.

After selecting the appropriate action and receiving feedback from that action (i.e., the selected base learner), the received feedback is evaluated, and the probability vector of the selected action is updated in the Q-table. The choice of base learners is probabilistic and calculated using the Boltzmann equation (Frankel 2019), which is defined as follows:

$$p(s,a) = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a' \in A} e^{\frac{Q(s,a')}{\tau}}} \tag{3}$$

where the parameter τ is the temperature, and it controls the randomness of the search. Determining the exact value of τ is very important because if the value of τ is not correctly selected, some part of the problem space may be ignored and never searched, and the algorithm will be stopped at local optima. If $\tau \to \infty$ the learning agent chooses its actions at random, and if $\tau \to 0$ the method of choice will be the greedy action selection. And if $\tau \in [0, \infty)$ the actions that have a higher probability of choice will have a higher chance of being selected than those with a lower probability. In the proposed method, the parameter τ is defined as a function of time in such a way that its value decreased over time.

This process, i.e., action selection and receiving feedback, continued as long as the action selection is converged to one of the selectable actions by the Q-algorithm, and the maximum amount of reward can be achieved. The details of the Stateless Q-learning component are shown in Algorithm 2. Details of the Q-learning component are shown in Fig. 6.

As shown in Fig. 6, the Q-table initialization step is optional. At this stage, the default knowledge can be transferred to the Q-table as the initial knowledge, and in another approach, the Q-table can be left empty so that in the learning process, its contents can be completed by receiving feedback from the environment. This option has been used to perform two different evaluations, one with prior knowledge and the other without prior knowledge, which is discussed in the evaluation section.

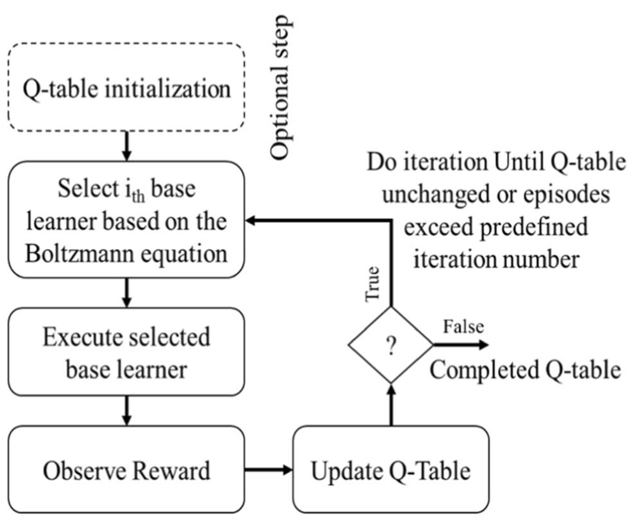


Fig. 6 Details of the Q-learning component

Choosing the appropriate base learner that can provide the most reward for the next stage is done using the Boltzmann equation. Then, the selected base learner is executed and receives the corresponding reward. After that, the Q-table cell that corresponds to the desired state/action is updated using Eq. (1). This process continues until the contents of the Q-table cells do not change significantly or until a predefined number of repetitions is reached. After completing the Q-table, it is used for the testing phase.

Algorithm 2 Stateless Q-learning component in the proposed method

Component Stateless Q-Learning

1: **Assumption**
2: $S_i : \{s_1, s_2, \ldots, s_m\}$ // *States:* input *samples*
3: $A : \{a_1, a_2, \ldots, a_n\}$ // *Actions: base learners*
4: $\gamma = 0.95$ // *Discount parameter*
5: $\tau = 1.0$ // max τ = *1.0, min* τ = *0.01*
6: $Q_t(a) \leftarrow 0$ // *Q-table initialization*
7: **Repeat** (**for each** *episode*)
8: **Initialize** s_i // *each input sample*
9: **Repeat** (**for each** *state*)
10: $a_j = \arg\max\{p(s_i, a_i) = e^{\frac{Q(s,a)}{\tau}} \Big/ \sum_{a' \in A} e^{\frac{Q(s,a')}{\tau}}\}$ // *select* i_{th} *base learner*
11: **Execute** *action* a_j
12: **Observe** $r(a_j)$ // *reward value*
13: $Q_{t+1}(a) = (1-\alpha)Q_t(a) + \alpha[r + \gamma \max_{a'} Q(a')]$ // *Update Q-Table*
14: **Scale down** τ
15: **Until** S is the *terminated state*
16: **Until** the *Q-table is unchanged or episodes exceed the predefined iteration number*

5 Evaluation

In this section, the data used in the evaluation are introduced and the experimental results are also presented. In this section, the data used in the evaluation are introduced and the experimental results are also presented. The details of the selected datasets are given in the first subsection and the details of the experimental result are given in the second subsection.

5.1 Datasets

To evaluate the proposed method, six data sets are used. The detail of these datasets is explained as follows.

Stanford—Sentiment 140 corpus. This dataset contains 1,600,000 tweets for sentiment analysis systems its instances are labeled with both positive and negative labels (Pang and Lee 2005). The average sentence length in this dataset is 12.8 words.

Large Dataset of Movie Reviews. This data collection includes 50,000 comments on cinematic films (Lin et al. 2011). These comments are organized in both positive and negative directions. The average sentence length in this dataset is 194.6 words.

Sentence Polarity Dataset v1.0. This data set (Pang et al. 2002) consists of processed 5331 positive samples and 5331 negative samples. The average sentence length in this dataset is 19.3 words.

Internet Movie Database. This data set (Dua and Graff 2017) consists of 1400 samples, of which 700 samples are labeled with a positive mark, and 700 samples are labeled with a negative mark. The average sentence length in this dataset is 650.1 words.

Yelp Review. This dataset (Dua and Graff 2017) contains 560,000 train samples and 38,000 test samples. The average sentence length in this dataset is 109.9 words.

Amazon Review. This data set (Dua and Graff 2017) con sists of 800,000 training samples and 200,000 testing samples. The average sentence length in this dataset is 70.5 words.

All of the above text data sets contain real opinions of individuals in different domains. Both the high volume of data and the diversity of their domain were the best options for their selection of our evaluation.

5.2 Experimental Results

To prove the performance of the proposed method, the evaluation is performed in two ways. In the first evaluation mode, Q-table is initially set to zero. Equal initialization means that the learning algorithm has no basic knowledge about the environment, and as noted, it is capable of dealing with problems with dynamic space.

In the second evaluation, the initial probability of choosing each learning algorithm's actions, i.e., the choice of each of the base learners, is set to non-equal and non-zero. The calculation of the values set for the initial probability of selection is shown in Fig. 7. Setting the values is based on how each base learner performs during the learning process. The values set for the initial probability of action selection, i.e. the Q-table initial values, are

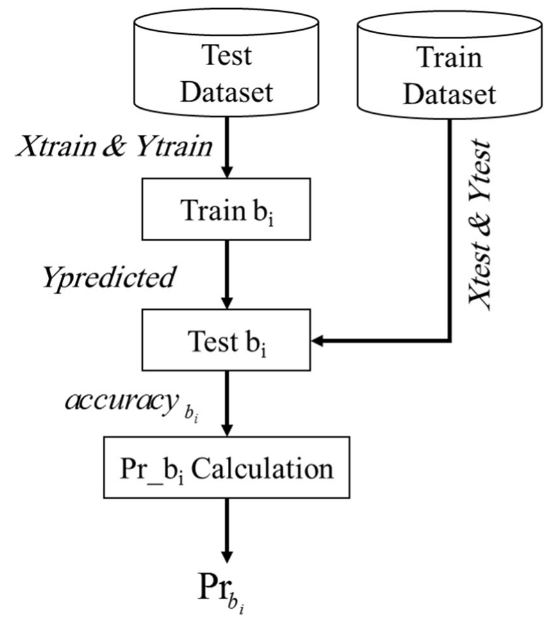


Fig. 7 Calculating the Q-table values in the second evaluation

calculated based on the calculations presented in Algorithm 3.

The values obtained from the calculations in Algorithm 3 are intended to adjust the initial value of the Q-table in the second evaluation. The evaluations were based on the six textual datasets introduced previously. In Fig. 8, the proposed method is compared with conventional methods in the literature on the subject of ensemble learning and the state of the art (Savargiv et al. 2022), (Savargiv et al. 2021).

As shown in Fig. 8, the proposed method provides higher accuracy performance in the absence of any prior knowledge of the environment. However, the results are unfavorable when the proposed method receives default knowledge of the environment. The high accuracy obtained in the proposed method confirms that it is domain-independent and updates its execution adaptively to achieve the highest possible reward in conditions where the environment has dynamic behavior. The selected datasets for evaluation are from different domains, and the proposed method has yielded higher accuracy in all of them. Although the proposed method is equipped with default knowledge of the environment, no satisfactory results have been obtained. The reason for this weakness in accuracy is that the proposed method operates consistently for all test samples that exhibit dynamic behavior and vary under different conditions. Therefore, the proposed method is unable to adapt to unexpected conditions.

Algorithm 3 Initial probability selection calculation

Function Initial Probability Selection

input
  $D_i = \{ D_1, D_2, \ldots, D_n \}$ *// Dataset containing data to be specified*
output
  $\Pr_{b_i} = \{\Pr_{b_1}, \Pr_{b_2}, \ldots, \Pr_{b_n}\}$ *// the probability of selection of base learners (actions)*
assumption
  $B_i = \{ b_1, b_2, \ldots, b_n \}$ *// The list of base learners in the ensemble*
for each b_i **in** ensemble **do**
    Training b_i
    Testing b_i
    Calculate $accuracy_{b_i} = TP_{b_i} + TN_{b_i} / TP_{b_i} + TN_{b_i} + FP_{b_i} + FN_{b_i}$
end **//for**
$\Pr_{b_i} = accuracy_{b_i} / \sum accuracy_{b_i}$

The comparison between the proposed method and the existing methods in the subject literature in terms of performance measures is shown in Table 6. The considered performance criteria are:

Precision: The exactness of the results is calculated by this criterion. Precision is calculated as the following:

$$percision(P) = \frac{TP}{TP + FP} \tag{4}$$

where TP (True Positive) means correctly labeled as positives, and FP (False Positive) means negative samples labeled as positive.

Recall: The completeness of the results is displayed by this criterion and it is calculated as the following:

$$\mathrm{Re}call(R) = \frac{TP}{TP + FN} \tag{5}$$

And FN (False Negative) means positive examples labeled as negative.

F1-score: This measurement is the harmonic mean of precision and recall and is calculated as:

$$F1 - score = \frac{2 * persicion * recall}{persicion + recall} \tag{6}$$

Accuracy: This criterion is calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

Table 6 reveals that the proposed method has the highest values in terms of exactness and completeness criteria, which indicates the high power of the proposed method compared to the traditional methods of ensemble learning

The emphasis of the proposed method is on improving efficiency from the point of view of accuracy. On the other hand, the proposed method has eliminated the need for basic knowledge. In Table 5 and 6, the time criterion has been examined. In this Table 5, the times required for preprocessing and BoW creation for 1000 samples are considered. The reason for this choice is to neutralize the number of samples in each dataset in evaluating the time criterion to provide the same conditions for all datasets. In the evaluation related to the time criterion, the length of each sample is considered equal to a 5000-dimensional feature vector, and 70% of the total number of samples is assigned to the training phase and 30% to the testing phase. In the examination of the time criterion, the time required for the training phase of base learners is the same for all methods, therefore, it has been avoided to mention this item in Table 7.

The evaluation was done on Python 3.6 environment and Core(TM)2 Duo 2.00GHz main processor and 4.00GB RAM on a 64-bit Operating System were used.

In Table 8, the time required for the test phase for each of the methods is shown for different datasets.

As can be seen in Table 8, since the proposed method is based on the idea of reinforcement learning, i.e. receiving feedback from the environment, it requires a lot of time for the classification process, however, the proposed method in comparison with similar methods, it shows a more favorable performance from the point of view of the accuracy criterion. The proposed method does not claim to improve efficiency in terms of time criteria. The results shown in Table 7 are the average results of twenty different evaluations.

The values considered for the parameters related to Q-learning have been based on the literature, and the setting best values for it are for the improvement activities of the proposed method that is outside the scope discussed in this article. To examine the proposed method more precisely, the results of evaluations of the six textual datasets have been re-examined. A re-examination of the results

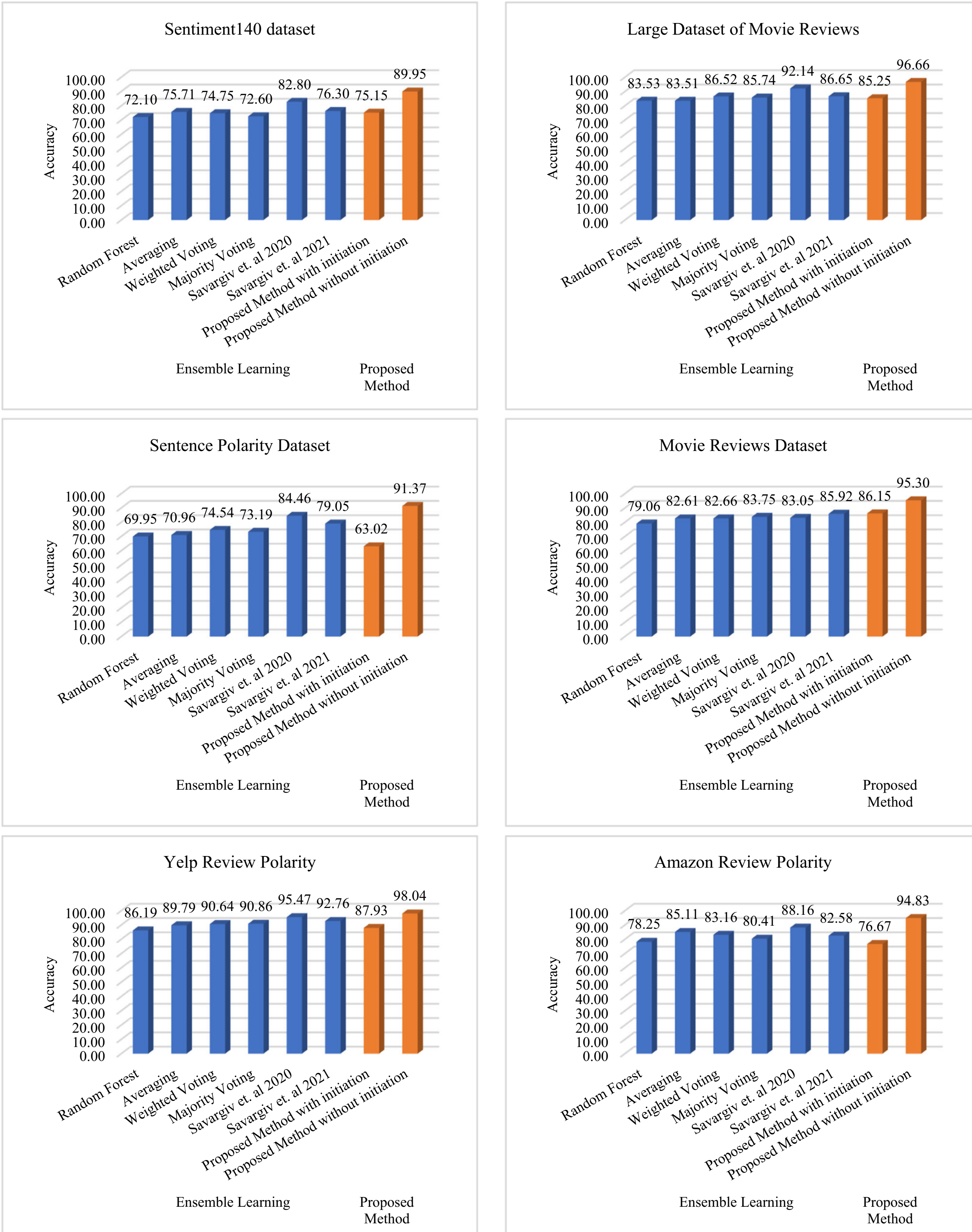


Fig. 8 Comparison between the proposed method and conventional aggregation methods in the ensemble learning

Table 6 Comparison of the proposed method with the existing methods of ensemble learning based on statistical metrics. P, R, and F1 refer to Precision, Recall, and F1-score

		Positive class			Negative class		
Sentiment140 dataset	Accuracy(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
Random forest	72.10	70.16	73.08	71.590	74.05	71.19	72.592
Averaging	75.71	78.46	74.61	76.487	72.95	76.94	74.892
Weighted voting	74.75	77.50	73.65	75.526	71.99	75.98	73.931
Majority voting	72.60	75.35	71.50	73.375	69.84	73.83	71.780
(Savargiv et al. 2022)	82.80	84.65	80.80	82.680	81.24	82.23	81.732
(Savargiv et al. 2021)	76.30	76.90	73.05	74.926	71.27	75.26	73.211
Proposed method with initiation	75.15	77.90	74.05	75.926	72.39	76.38	74.331
Proposed method without initiation	89.95	90.02	89.93	89.975	89.88	89.97	89.925
Large dataset of movie reviews							
Random forest	83.53	86.12	81.78	83.894	80.98	85.48	83.169
Averaging	83.51	88.32	80.28	84.108	78.75	87.40	82.850
Weighted voting	86.52	91.33	83.29	87.125	81.76	90.41	85.868
Majority voting	85.74	90.55	82.51	86.343	80.98	89.63	85.086
(Savargiv et al. 2022)	92.14	91.64	83.60	87.436	82.95	91.60	87.061
(Savargiv et al. 2021)	86.65	87.36	79.32	83.146	86.96	87.59	87.274
Proposed method with initiation	85.25	85.74	84.79	85.262	84.79	85.73	85.257
Proposed method without initiation	96.66	97.15	96.20	96.673	96.20	97.14	96.668
Sentence polarity dataset							
Random forest	69.95	75.20	68.67	71.787	64.65	71.57	67.934
Averaging	70.96	77.48	68.84	72.905	65.58	75.13	70.031
Weighted voting	74.54	81.06	72.42	76.497	67.81	77.36	72.271
Majority voting	73.19	79.71	71.07	75.142	66.46	76.01	70.915
(Savargiv et al. 2022)	84.46	89.63	80.99	85.091	76.39	85.94	80.884
(Savargiv et al. 2021)	79.05	85.57	76.93	81.020	63.49	73.04	67.931
Proposed method with initiation	63.02	64.49	62.01	63.226	61.42	64.04	62.703
Proposed method without initiation	91.37	92.84	90.36	91.583	89.77	92.39	91.061
Movie reviews dataset							
Random forest	79.06	73.68	80.99	77.162	84.03	77.56	80.665
Averaging	82.61	89.09	77.81	83.069	76.64	88.46	82.127
Weighted voting	82.66	89.14	77.86	83.119	76.69	88.51	82.177
Majority voting	83.75	90.23	78.95	84.214	77.78	89.60	83.273
(Savargiv et al. 2022)	83.05	83.09	72.11	77.212	70.30	81.67	75.560
(Savargiv et al. 2021)	85.92	80.85	81.29	81.069	73.00	84.37	78.274
Proposed method with initiation	86.15	85.59	86.30	85.944	86.68	86.02	86.349
Proposed method without initiation	95.30	94.74	95.45	95.094	95.83	95.17	95.499
Yelp review polarity							
Random forest	86.19	82.19	89.70	85.781	90.31	83.16	86.588
Averaging	89.79	93.35	87.28	90.213	86.15	92.77	89.338
Weighted voting	90.64	94.20	88.13	91.064	87.00	93.62	90.189
Majority voting	90.86	94.42	88.35	91.284	87.22	93.84	90.409
(Savargiv et al. 2022)	95.47	99.25	92.98	96.013	91.92	98.21	94.961
(Savargiv et al. 2021)	92.76	96.05	90.78	93.341	88.90	95.19	91.938
Proposed method with initiation	87.93	88.60	87.35	87.971	87.25	88.55	87.895
Proposed method without initiation	98.04	98.71	97.46	98.081	97.36	98.66	98.006
Amazon review polarity							
Random forest	78.25	79.26	77.58	78.411	77.24	78.95	78.086

Table 6 (continued)

Sentiment140 dataset	Accuracy(%)	Positive class			Negative class		
		P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
Averaging	85.11	93.33	80.74	86.580	76.96	91.18	83.469
Weighted voting	83.16	91.38	78.79	84.619	75.01	89.23	81.504
Majority voting	80.41	88.63	76.04	81.854	72.26	86.48	78.733
(Savargiv et al. 2022)	88.16	94.13	82.10	87.704	84.29	95.26	89.440
(Savargiv et al. 2021)	82.58	81.10	83.65	82.355	79.59	93.22	85.867
Proposed method with initiation	76.67	77.83	75.63	76.714	75.53	77.76	76.629
Proposed method without initiation	94.83	95.99	93.79	94.877	93.69	95.92	94.792

The items displayed in bold in these two tables are to better display the achievements of the proposed method in this paper

Table 7 Statistical information related to the datasets, including the time required for pre-processing, the time required for BoW creation, and the average length of the text samples

Dataset	# instance	Average length of a Sentence (word)	pre-processing time (ms)	BoW creation time(ms)
Stanford—sentiment 140 corpus	1,600,000	12.8	105.3	18.5
Large dataset of movie reviews	50,000	194.6	658.6	53.4
Sentence polarity dataset v1.0	10,600	19.3	93.4	10.7
Internet movie database	1,400	650.1	2120.6	229.4
Yelp review	560,000	109.9	579.5	46.9
Amazon review	800,000	70.5	251.8	22.4

Table 8 Time evaluation for the test phase in milliseconds

Dataset/method	Random forest	Averaging	Weighted voting	Majority voting	Savargiv et. al 2020	Savargiv et. al 2021	Proposed method
Stanford—sentiment 140 corpus	40.6	50.4	49.5	52.7	62.1	65.3	63.3
Large dataset of movie reviews	83.2	113.8	124.3	114.6	146.8	150.3	148.7
Sentence polarity dataset v1.0	43.3	50.5	39.0	38.6	23.5	46.0	56.6
Internet movie database	299.6	398.4	421.8	400.6	476.3	508.6	514.9
Yelp review	72.9	105.6	117.5	106.4	138.2	147.4	153.4
Amazon review	51.2	75.7	66.4	62.1	66.1	71.9	89.6

was performed using the Friedman test statistical verification (López-Vázquez and Hochsztain 2019). This evaluation is based on statistics and offers two types of rankings. The distance between the rows indicates the amount of improvement obtained by different methods. The results of the Friedman test statistical verification are shown in Table 9.

The results obtained from Friedman test statistical verification reveal a significant difference between the proposed method and the existing method in the literature of ensemble learning. The difference between the proposed method and the other methods is found in the mean rank criterion, which is a strong argument for the improvement obtained by the proposed method.

Since the proposed method has been evaluated on six textual datasets with different domains, and a high volume of data, the results of the studies are valid and reliable. On the other hand, by performing evaluations on datasets from different domains, the high accuracy criteria results show that the proposed method is domain-independent.

Typically, as data volume increases, the performance of data mining algorithms declines. In the proposed method, which is a novel way of integrating different base learners, this challenge is resolved, and the efficiency of the

Table 9 The Friedman ranking

Method	Mean rank	Final rank
Proposed method without initiation	8	1
(Savargiv et al. 2022)	6.5	2
(Savargiv et al. 2021)	5.67	3
Weighted voting	4.17	4
Majority voting	3.83	5
Averaging	3.33	6
Proposed method with initiation	3	7
Random forest	1.5	8

proposed method does not diminish in the face of high dimension data. The reason for this strength is the use of reinforcement learning ideas in the challenge of integrating base learners in the ensemble learning approach. The significant results obtained from the evaluations of the mentioned high-volume datasets in the previous section confirmed this claim. The proposed method has the potential to address high-volume issues.

6 Conclusion and Future Work

This paper proposes a new method for integrating base learners into the ensemble learning approach based on the reinforcement learning idea. The proposed method dynamically aggregates base learners after receiving feedback from the environment using the Q-learning algorithm. Since Q-learning does not require prior knowledge of the environment, it is well-suited for dealing with dynamic environments. To evaluate the proposed method, text data, which exhibits diverse behavior across different domains, was selected. The proposed method's adaptability through reinforcement learning eliminates the need for predefined polarity dictionaries in text mining, achieving high accuracy without requiring prior knowledge of the environment. The proposed method also addresses the data-driven limitations of data mining algorithms and can handle high-dimensional problems using reinforcement learning.

The proposed method is a novel approach to integrating classifiers in an ensemble structure and was evaluated in the field of sentiment analysis. The method is language-independent, but appropriate preprocessing is necessary for sentiment analysis in different languages. Preprocessing is a general part of every process, and the proposed method focuses on how to combine classifiers, without making any claims regarding preprocessing. While the proposed method was evaluated on text data, it can be applied to integrating classifiers in any area of expertise.

In ensemble learning techniques, the training phase must be completed for all base learners before grouping them. The time required for training base learners is the same regardless of the integration method used. However, the process of combining base learners will differ between classical and reinforcement learning-based techniques. Classical methods such as simple voting, majority voting, and averaging do not require significant time to create an ensemble, while reinforcement learning-based techniques need more time due to the need for feedback from the environment and repetitive training phases. The proposed method is no exception, requiring more time than classical methods. Therefore, this study does not claim to improve processing time in the proposed method.

In future work, the authors intend to tune the optimal values for Q-learning parameters and select the most appropriate action using Q-learning.

Tuning the optimal values for Q-learning parameters and selecting the most appropriate action by Q-learning is among the tasks that the authors intend to do in the future.

Declarations

Conflict of interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abasabadi S, Nematzadeh H, Motameni H, Akbari E (2021) Automatic ensemble feature selection using fast non-dominated sorting. Inf Syst. https://doi.org/10.1016/j.is.2021.101760

Ahmed M, Chen Q, Li Z (2020) Constructing domain-dependent sentiment dictionary for sentiment analysis. Neural Comput Appl. https://doi.org/10.1007/s00521-020-04824-8

Aram KY, Lam SS, Khasawneh MT (2023) Cost-sensitive max-margin feature selection for SVM using alternated sorting method genetic algorithm. Knowl-Based Syst 267:110421

Banik R, Biswas A (2023) Improving solar PV prediction performance with RF-catboost ensemble: a robust and complementary approach. Renewable Energy Focus. https://doi.org/10.1016/j.ref.2023.06.009

Barrachina-Muñoz S, Chiumento A, Bellalta B (2021) Stateless reinforcement learning for multi-agent systems: the case of spectrum allocation in dynamic channel bonding WLANs. In: 2021 Wireless Days (WD). pp 1–5

Basha SM, Rajput DS (2019) A roadmap towards implementing parallel aspect level sentiment analysis. Multimed Tools Appl 78:29463–29492

Beigi OM, Moattar MH (2021) Automatic construction of domain-specific sentiment lexicon for unsupervised domain adaptation and sentiment classification. Knowledge-Based Syst 213:106423

Chaturvedi I, Cambria E, Welsch RE, Herrera F (2018) Distinguishing between facts and opinions for sentiment analysis: Survey and challenges. Inf Fusion 44:65–77

Chen S, Devraj AM, Lu F, Busic A, Meyn S (2020) Zap Q-Learning with nonlinear function approximation. Adv Neural Inf Process Syst 33:16879–16890

Cherubin G (2019) Majority vote ensembles of conformal predictors. Mach Learn 108:475–488

Chiong R, Fan Z, Hu Z, et al (2018) A sentiment analysis-based machine learning approach for financial market prediction via news disclosures. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion. pp 278–279

Coletta LFS, da Silva NFF, Hruschka ER, Hruschka ER (2014) Combining classification and clustering for tweet sentiment analysis. In: 2014 Brazilian Conference on Intelligent Systems. pp 210–215

Da'u A, Salim N, Rabiu I, Osman A (2020) Weighted aspect-based opinion mining using deep learning for recommender system. Expert Syst Appl 140:112871

Dang NC, Moreno-Garcia MN, la Prieta F (2020) Sentiment analysis based on deep learning: a comparative study. Electronics 9:483

Deegalla S, Walgama K et al (2022) Random subspace and random projection nearest neighbor ensembles for high dimensional data. Expert Syst Appl 191:116078

Dua D, Graff C, others (2017) UCI machine learning repository. http://archive.ics.uci.edu/ml. Accessed 12 Sept 2019

Frankel A (2019) Solution of the Generalized Linear Boltzmann Equation for Transport in Multidimensional Stochastic Media. arXiv Preprint. arXiv:190105874

Gargiulo F, Silvestri S, Ciampi M, De Pietro G (2019) Deep neural network for hierarchical extreme multi-label text classification. Appl Soft Comput 79:125–138

Hajewski J, Oliveira S (2019) Distributed SmSVM ensemble learning. In: INNS big data and deep learning conference. pp 7–16

Hernández-Fernández A, Mora E, Hernández MIV (2019) When a new technological product launching fails: a multi-method approach of facial recognition and E-WOM sentiment analysis. Physiol Behav 200:130–138

Huddar MG, Sannakki SS, Rajpurohit VS (2020) Multi-level feature optimization and multimodal contextual fusion for sentiment analysis and emotion classification. Comput Intell 36:861–881

Jiang J, Yu Y, Wang Z et al (2019) Ensemble super-resolution with a reference dataset. IEEE Trans Cybern 50:4694–4708

Ju BS, Kwag S, Lee S (2023) Performance-based drift prediction of reinforced concrete shear wall using bagging ensemble method. Nucl Eng Technol 55:2747–2756

Kaur RP, Kumar M, Jindal MK (2019) Newspaper text recognition of Gurumukhi script using random forest classifier. Multimed Tools Appl 79(11):7435–7448

Kaya B (2020) A hotel recommendation system based on customer location: a link prediction approach. Multimed Tools Appl 79:1745–1758

Kumar V, Pujari AK, Padmanabhan V et al (2018) Multi-label classification using hierarchical embedding. Expert Syst Appl 91:263–269

Kumar P, Gupta GP, Tripathi R (2021) A distributed ensemble design based intrusion detection system using fog computing to protect the internet of things networks. J Ambient Intell Human Comput 12(10):9555–9572

Li X, Wu P, Wang W (2020) Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong. Inf Process Manag 57(5):102212

Liang D, Fan G, Lin G, et al (2019) Three-stream convolutional neural network with multi-task and ensemble learning for 3d action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. p 0

Lin D, Matsumoto Y, Mihalcea R (2011) Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies. Association for Computational Linguistics, Portland, Oregon, USA

López-Vázquez C, Hochsztain E (2019) Extended and updated tables for the Friedman rank test. Commun Stat Methods 48:268–281

Ma X, Zeng J, Peng L et al (2019) Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis. Futur Gener Comput Syst 93:304–311

Madasu A, Elango S (2020) Efficient feature selection techniques for sentiment analysis. Multimed Tools Appl 79:6313–6335

Menor-Flores M, Vega-Rodríguez MA (2023) Boosting-based ensemble of global network aligners for PPI network alignment. Expert Syst Appl 230:120671

Morente-Molinera JA, Kou G, Pang C et al (2019) An automatic procedure to create fuzzy ontologies from users' opinions using sentiment analysis procedures and multi-granular fuzzy linguistic modelling methods. Inf Sci (ny) 476:222–238

Munezero M, Montero CS, Sutinen E, Pajunen J (2014) Are they different? Affect, feeling, emotion, sentiment, and opinion detection in text. IEEE Trans Affect Comput 5:101–111

Pang B, Lee L (2005) Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. arXiv Preprint. arXiv:cs/0506075

Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? Sentiment classification using machine learning techniques. arXiv preprint. arXiv:cs/0205070

Pham D-H, Le A-C (2018) Learning multiple layers of knowledge representation for aspect based sentiment analysis. Data Knowl Eng 114:26–39

Priya S, Uthra RA (2021) Retracted article: comprehensive analysis for class imbalance data with concept drift using ensemble based classification. J Ambient Intell Human Comput 12(5):4943–4956

Sagi O, Rokach L (2018) Ensemble learning: a survey. Wiley Interdiscip Rev Data Min Knowl Discov 8:e1249

Sánchez-Núñez P, de las Heras-Pedrosa C, Peláez JI (2020) Opinion mining and sentiment analysis in marketing communications: a science mapping analysis in web of science (1998–2018). Soc Sci 9:23

Savargiv M, Masoumi B, Keyvanpour MR (2021) A New random forest algorithm based on learning automata. Comput Intell Neurosci 2021:1–19. https://doi.org/10.1155/2021/5572781

Savargiv M, Masoumi B, Keyvanpour MR (2022) A new ensemble learning method based on learning automata. J Ambient Intell Humaniz Comput 13:3467–3482. https://doi.org/10.1007/s12652-020-01882-7

Sharma U, Datta RK, Pabreja K (2020) Sentiment analysis and prediction of election results 2018. In: Social Networking and Computational Intelligence. Springer, pp 727–739

Shaw R, Howley E, Barrett E (2021) Applying reinforcement learning towards automating energy efficient virtual machine consolidation in cloud data centers. Inf Syst. https://doi.org/10.1016/j.is.2021.101722

Sisodia DS, Bhandari S, Reddy NK, Pujahari A (2020) A comparative performance study of machine learning algorithms for sentiment analysis of movie viewers using open reviews. Perform Manag Integr Syst Appl Softw Eng. https://doi.org/10.1007/978-981-13-8253-6_10

Song W, Wen Z, Xiao Z, Park SC (2021) Semantics perception and refinement network for aspect-based sentiment analysis. Knowledge-Based Syst 214:106755

Sutton RS, Barto AG (2018) Reinforcement learning: an introduction. MIT press, Cambridge

Vidhya R, Vadivu G (2021) Retracted article: towards developing an ensemble based two-level studentclassification model (ESCM) using advanced learning patterns and analytics. J Ambient Intell Human Comput 12(7):7095–7105

Wang X, Yan WQ (2020) Cross-view gait recognition through ensemble learning. Neural Comput Appl 32:7275–7287

Werbin-Ofir H, Dery L, Shmueli E (2019) Beyond majority: label ranking ensembles based on voting rules. Expert Syst Appl 136:50–61

Wilhelmi F, Bellalta B, Cano C, Jonsson A (2017) Implications of decentralized Q-learning resource allocation in wireless networks. In: 2017 ieee 28th annual international symposium on personal, indoor, and mobile radio communications (pimrc). pp 1–5

Wu C, Xiong Q, Yi H et al (2021) Multiple-element joint detection for aspect-based sentiment analysis. Knowledge-Based Syst 223:107073

Zhou H, Song F (2015) Aspect-level sentiment analysis based on a generalized probabilistic topic and syntax model. In: The Twenty-Eighth International Flairs Conference