

An Ensemble Classifier for Stock Trend Prediction Using Sentence-Level Chinese News Sentiment and Technical Indicators

Chun-Hao Chen^{1*}, Po-Yeh Chen², Jerry Chun-Wei Lin^{3*}

¹ Department of Information and Finance Management, National Taipei University of Technology, Taipei (Taiwan)

² Department of Computer Science and Information Engineering, Tamkang University, Taipei (Taiwan)

³ Department of Computer Science, Electrical Engineering, and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen (Norway)

Received 22 November 2021 | Accepted 28 January 2022 | Published 11 February 2022



ABSTRACT

In the financial market, predicting stock trends based on stock market news is a challenging task, and researchers are devoted to developing forecasting models. From the existing literature, the performance of the forecasting model is better when news sentiment and technical analysis are considered than when only one of them is used. However, analyzing news sentiment for trend forecasting is a difficult task, especially for Chinese news, because it is unstructured data and extracting the most important features is difficult. Moreover, positive or negative news does not always affect stock prices in a certain way. Therefore, in this paper, we propose an approach to build an ensemble classifier using sentiment in Chinese news at sentence level and technical indicators to predict stock trends. In the training stages, we first divide each news item into a set of sentences. TextRank and word2vec are then used to generate a predefined number of key sentences. The sentiment scores of these key sentences are computed using the given financial lexicon. The sentiment values of the key phrases, the three values of the technical indicators and the stock trend label are merged as a training instance. Based on the sentiment values of the key sets, the corpora are divided into positive and negative news datasets. The two datasets formed are then used to build positive and negative stock trend prediction models using the support vector machine. To increase the reliability of the prediction model, a third classifier is created using the Bollinger Bands. These three classifiers are combined to form an ensemble classifier. In the testing phase, a voting mechanism is used with the trained ensemble classifier to make the final decision based on the trading signals generated by the three classifiers. Finally, experiments were conducted on five years of news and stock prices of one company to show the effectiveness of the proposed approach, and results show that the accuracy and P / L ratio of the proposed approach are 61% and 4.0821 are better than the existing approach.

KEYWORDS

Chinese News Analysis, Sentiment Analysis, Stock Trend Prediction, Support Vector Machine, Technique Indicators.

DOI: 10.9781/ijimai.2022.02.004

I. INTRODUCTION

FINANCIAL data analysis is an attractive research area for researchers and many topics could be studied, e.g., portfolio management and optimization [1]–[3], trading strategy discovery and optimization [4], [5], news sentiment analysis [6], [7]. One of the challenging research topics is the prediction of stock trends [8]–[10]. For example, Chen et al. proposed a graph feature-based neural network model that uses constructed stock networks, the corresponding feature matrices, trading data, and technical indicators to predict stock trends [8]. Considering trading transaction data, Long et al. proposed a framework that merges trading and market information for stock prediction using an attention-based bidirectional network for long-term memory [10].

The literature shows that financial news sentiment has an impact on stock prices [11], [12]. Through experiments with historical news articles of Hong Kong stock market, Li et al. found that the prediction models with sentiment analysis are better than the bag-of-words model in validation and test data set [11]. Loeffler et al. analyzed the tone of voice in Moody's rating reports and pointed out that the short-term effects of tone of voice in reports with negative ratings affect investors' mood and attention differently [12]. To determine the impact of news sentiment on financial returns, Kelly et al. found that annual returns are better when news sentiment is considered in a trading strategy than the buy-and-hold strategy for the stock and oil markets [13].

For predicting stock trends, several approaches have been presented to solve this problem using different techniques, including neural networks [8], [9], [14], rough set [15], support vector machine (SVM) [10], genetic algorithms (GA) [16] and others [17]. Many algorithms that consider news sentiment have also been proposed for predicting stock trends [18]–[20]. For example, Hao et al. proposed a fuzzy twin

* Corresponding author.

E-mail addresses: chchen@ntut.edu.tw (C. H. Chen), jerrylin@ieee.org (J. C.W. Lin).

support vector machine based on the hidden topic model and the emotional information of news to predict stock trends [18]. Wen et al. presented an SVM model for predicting stock price movements that takes into account not only the content of news but also the information hidden in the relationships between news [19].

From the literature, we have seen that there are still two problems that need to be considered in order to build a more practical predictive model: (1) The first problem is that the release of positive news does not mean that the stock price will increase in the next trading days. On the contrary, negative news does not always mean that the stock price will fall. Therefore, the first problem to solve is how to use the sentiment from financial news to build multiple forecasting models for trend prediction. (2) To extract sentiment from the corpus, a common method is to search for keywords from the entire corpus and then calculate the sentiment value based on the generated keywords. However, searching keywords from key phrases might provide a better representative set of keywords than those from the whole corpus. So the second question is how to generate the key phrases and find a representative set of keywords for the classification attributes from them.

To address the aforementioned problems, we propose a three-stage algorithm to create an ensemble classifier that uses sentiment in Chinese news at the sentence level and technical indicators to predict stock trends. In the data preprocessing stage, each news item is used to find a set of keywords using TextRank, and divided into a set of sentences. Then, the keywords generated from the sentences are compared with the keywords from the corpus using word2vec to find the predefined number of key sentences. The sentiment values of the generated key phrases are calculated using the given financial lexicon. The sentiment values of the key phrases and the news title, as well as the two technical indicators, are used as classification attributes along with the trend label generated from the difference between the closing and opening prices on the news release day to form a training instance. Based on the sentiment values of the key records, the training instance is stored in the training datasets for positive and negative news. In the modeling phase, the two datasets are used to build models for positive and negative stock trend predictions using the Support Vector Machine (SVM). To increase the reliability of the generated trading signals, a classifier based on the Bollinger bands is also created. These three classifiers are combined to form an ensemble classifier. In the prediction phase, the constructed ensemble classifier is used to predict trading signals for trading. To demonstrate the merits of the proposed approach, experiments were conducted on real financial news datasets. The contributions of this work are listed as follows:

- This paper proposes a three-phase framework for predicting stock trends using an ensemble classifier that incorporates news sentiment analysis and technical indicators.
- Using sentence-level sentiment analysis to build models for predicting positive and negative stock trends outperforms using a keyword-based approach in terms of returns.
- We also find that the news release day is highly associated with stock prices on the next trading day.

The remaining sections of this paper are organized as follows: Section II reviews the relevant literature. Section III explains the proposed framework and algorithm. In Section IV, experimental evaluations are given and discussed. Finally, Section V describes the conclusions and future work.

II. LITERATURE REVIEW

In this section, we first introduce related approaches to predicting stock trends in Section A. Then, related approaches to corpus analysis are described in Section B.

A. Stock Trend Prediction

The prediction of stock trends for investment is a challenging and widely studied research topic, as several factors need to be considered simultaneously when building the predictive model. Currently, there are several approaches to solve this problem using different techniques, including neural networks [8], [9], [14], rough set [15], support vector machine [10], genetic algorithms [16], and others [17], [21]. For example, Chen et al. proposed a Graph Convolutional Features based Convolutional Neural Network (GC-CNN) model for predicting stock trends [8]. It used the correlations and characteristics of stocks to generate stock market information. Then, trading data and technical indicators were used to observe stock information. Then, the stock market information and stock information were converted into images, which served as a training dataset for generating GC-CNN. Zhang et al. proposed an approach, called the status box method, to predict the change direction of stock prices [10]. It packs a period of stock points into three kinds of boxes to represent the stock status. Then, the specific feature extraction approach was developed to derive characteristic attributes. Finally, AdaBoost algorithm, Support Vector Machine and genetic algorithm were integrated to construct the classifiers to predict the stock price performance. Using keywords from financial news and technical indicators as classification attributes, Chen et al. proposed a genetic-based model to predict the stock trend using Support Vector Machine and genetic algorithm [16]. In this approach, keywords were first extracted from financial news. Then, more meaningful keywords were generated by combining 2 words. Feature selection was used to keep important keywords. The classifier was created based on the generated dataset using the support vector machine. To reduce the risk of trading, the technical indicators were used to increase the reliability of the trading signal proposed by the classifier. Since determining the hyperparameters of the predictive model and the technical indicators used is an optimization task, the genetic algorithm was used to find the appropriate hyperparameters. To deal with the profit bias in the model, Liu et al. proposed an effective metric called mean profit rate (MPR), which can be used to measure the return of the model and the correlation between the metric values to evaluate the stock trend prediction classifier [17]. Chen et al. proposed an approach for finding trading signals using long short-term memory neural network (LSTM) and genetic algorithms (GA) [22]. LSTM was used to learn fluctuations of stock prices and GA was then employed to find trading signals. Besides, the Kelly criterion was also employed to calculate optimal investment score for minimizing losses and maximizing returns. Take leading indicators into consideration, Wu et al. proposed the long short-term memory with leading indicators (LSTMLI) for prediction and showed that the prediction error can be reduced [23], and the LSTM-GA approach was also proposed based on LSTMLI [24]. In addition, based on the noisy equity state representation, Huang et al. proposed an algorithm for stock prediction using the recurrent neural network [25].

Since the literature shows that news sentiment has an impact on investors and investment goals [11], [12], many algorithms have also been proposed for predicting stock trends [18], [19]. For example, Long et al. proposed the semantic and structural (S&S) kernel for Support Vector Machine to predict the stock trend, considering not only the content of the news but also the information structures between the news [19]. The text point graph and keyword graph were first created from the given financial news. The S&S kernel was then defined over the two graphs and used to build the predictive model. To deal with and reduce the influence of outliers, Hao et al. proposed a fuzzy twin support vector machine classifier that used the emotional information and hidden topic model from the financial news to predict stock trends [18]. First, the high-level expression features were captured from the given financial news, including keywords, topic and sentiment

distribution vectors. Then, the labels for the training instances were generated from the stock price data. Finally, the fuzzy twin support vector machine classifier was created using the prepared training instances.

Since investor sentiment can also have an impact on the stock market, many algorithms have been proposed to predict market trends [26], [27]. Using user-generated online content on the stock news board, Li et al. compared different classifiers that considered investor sentiment in predicting stock prices and found that the long-term memory model had higher predictive power than others [27]. In addition, Derakhshan et al. proposed a graphical part-of-speech model to extract user opinions based on social network datasets to predict stock price movements [26].

B. Corpus Analysis

This section describes the relevant approaches to corpus analysis. First, the approaches for different types of applications are discussed, e.g., short text sentiment [28], [29], Chinese corpus analysis [30]–[32]. Based on the probabilistic linguistic term sets (PLTSs) and relevance theory, Song et al. proposed a framework for analyzing sentiment in short texts [28]. They designed a text representation model, called Word2PLTS, which uses PLTSs and relevance theory. Using Word2PLTS, each word was converted into a vector to show the possibilities for sentiment polarity. Finally, the sentiment and polarity classification system was developed using support vector machines for sentiment analysis of short texts.

For sentiment analysis in Chinese microblogs, Wu et al. first created several sentiment dictionaries, including original sentiment, emoji, and other dictionaries, to increase the coverage of sentiment words in Chinese microblogs. To improve the accuracy of sentiment analysis of Chinese microblogs, the semantic rule sets were further derived to represent the possible sentence patterns and information between sentences [29]. Wang et al. proposed a mixed character-word architecture using the semantic information of the intra-word characters to solve the problem of compositional Chinese sentence representation [32]. Two main strategies were employed to achieve the goal. The first uses the mask gate on characters to observe the relationship between them in a word. The second one applied the max-pooling operation on the words to find an optimal mixture of the compositional Chinese sentences. The results showed that the architecture was better than the character-based and word-based models. Wang et al. proposed an algorithm to observe the opinion orientation of Chinese news based on the word embedding and syntax rules [31]. To determine the orientation value of words, the cosine similarity between words and sentiment dictionaries was calculated using word2vec. The generated word vectors were then used to extract key phrases. The syntax rules and word vector similarity were used to analyze the alignment of the document based on the key phrases. Wang et al. proposed a Chinese parsing approach that uses the semantic string-matching sliding agreement (SMOSS) [30]. Their approach is divided into a training phase and a parsing phase. In the training phase, the tree node keywords were encoded in a treebank according to the semantic codes in the synonym dictionary. Then, the semantic templates were extracted from these tree nodes using sliding windows to build the semantic template library. In the parsing phase, the built semantic template library was used to extract the semantic code strings for better syntax parsing performance.

In recent years, many deep learning based approaches have also been proposed for analyzing sentiment in the corpus, including neural networks [33], [34], bidirectional long-term memory (BiLSTM) [35], recurrent neural networks (RNN) [36], the bidirectional encoder representations of transformers (BERT) [37], and reinforcement learning [38]. To extract sentiment polarities from a small training

corpus incorporating external knowledge, Chen et al. proposed a framework for sentiment analysis of Chinese reviews using a neural network [33]. In their approach, context features were extracted from review sentences. Then, the aspect-opinion pairs and their sentiment polarities were retrieved from the given sentiment knowledge graph as external knowledge. By using the generated training dataset, the created model was able to provide better results in analyzing sentiment from the limited review dataset. In order to consider the semantic and sentimental information of words to capture the contextual information, Xu et al. proposed an improved word representation model that integrates the sentimental information into the TF-IDF method to generate the word vectors [35]. Using the word representation model, a classifier was constructed using bidirectional long-term memory (BiLSTM) based on the generated working vectors to obtain effective sentiment analysis results. In order to provide producers with a useful tool to learn about consumers' needs and related aspects of online platforms, Yang et al. first defined a task called multi-entity aspect-based sentiment analysis (ME-ABSA) and proposed a method considering context, entity, and aspect memory to construct the classifier using a neural network [36]. In their approach, the network consisted of three layers, namely the interaction layer, the positional attention layer, and the RNN with attention layer, to form a context memory. The created model was then used to predict the sentiment polarity. To observe the implicit sentiments and the contagion process, Daudert proposed a sentiment analysis approach using a customized feed-forward neural network [34]. Based on different types of data structures, text preprocessing was performed. Then, the relationships between the records in the corpus were represented using a graph structure. In the graph, the features and constraints were used to form the vertices and edges between the vertices. The transformer-based model was used to extract more textual information from the corpus. The derived graph and text information were then used to construct the adapted feed-forward neural network. To analyze the sentiment of tweets, Pota et al. proposed an approach using the bidirectional encoder representations of transformers (BERT) [37]. In their approach, the data were first preprocessed to remove noise data. In addition, they attempted to utilize information hidden in the tweets, including emojis, hashtags, etc. The processed instances were then used to build the model using BERT. Ma et al. proposed a phonetically enriched text representation using reinforcement learning for Chinese sentiment analysis [38]. The main concept of their approach was to consider the two features, deep phonemic orthography and intonation variations, and fuse them with textual information to obtain an effective Chinese sentiment analysis. As a result, an algorithm called Disambiguate intonation for sentiment analysis was proposed using reinforcement learning. They also pointed out that the performance of their approach was better than the character-level approaches.

III. PROPOSED ENSEMBLE CLASSIFIER FOR STOCK PREDICTION ALGORITHM

In this section, we describe the proposed approach to build an ensemble classifier for predicting stock trends based on Chinese news sentiment and sentence-level technical indicators. In the following, Section A describes the three-phase framework of the proposed approach. The three phases of the proposed approach, including the preprocessing of data, the construction of the ensemble classifier, and the prediction phase, are described in Sections B, C, and D.

A. Flowchart of Proposed Approach

The three-stage framework of the proposed approach for building the ensemble classifier for predicting stock trends, incorporating sentence-level sentiment analysis of Chinese news and technical indicators, is shown in Fig. 1.

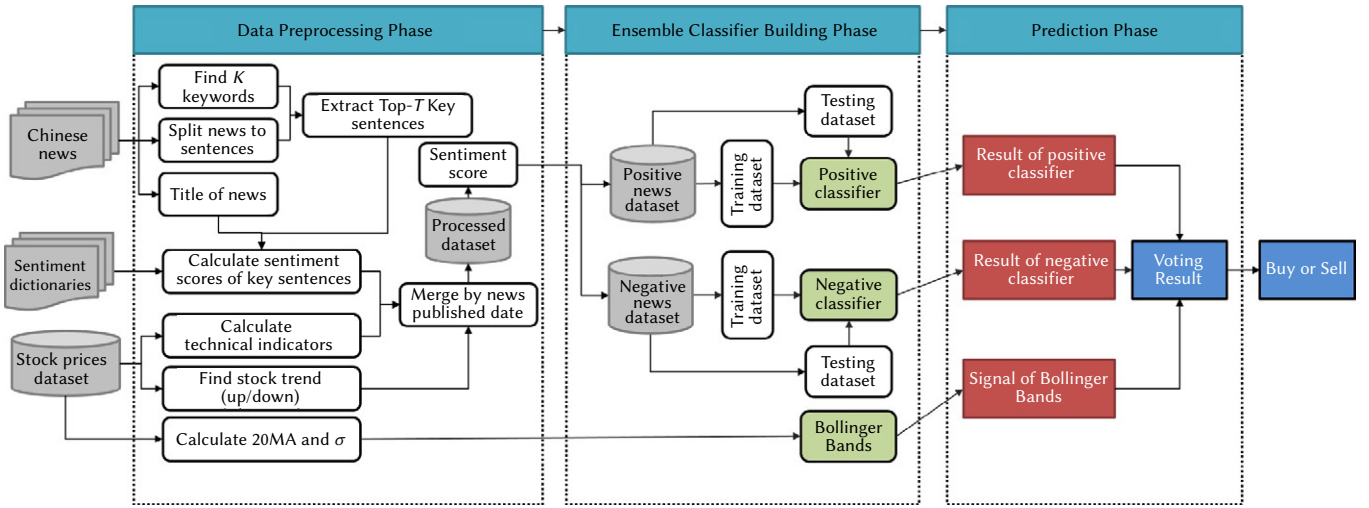


Fig. 1. Three-phase framework of the proposed approach.

Algorithm 1: Data preprocessing

Input: Chinese News $N = \{n_1, \dots, n_s\}$, stock prices dataset $S = \{s_1, \dots, s_j\}$, sentiment dictionaries $Dict^A = \{dict^P, dict^N\}$, number of keywords K , number of top key sentences T , and stock trend period P .

Output: Positive training dataset $posTrainDataset$, and negative training dataset $negTrainDataset$.

```

1  $posTrainDataset \leftarrow null$ ;
2  $negTrainDataset \leftarrow null$ ;
3 for data  $\in N$  do
4    $title, content, publishedDate \leftarrow getRelatedInformation(data)$ ;
5    $newsKeywords \leftarrow generateNewsKeywords(content, K)$ ;
6    $sentenceSet \leftarrow splitNewsToSentences(content)$ ;
7    $wordSentenceSet \leftarrow splitSentenceToWords(sentenceSet)$ ;
8    $WIS \leftarrow calculateWordImportanceScore(wordSentenceSet, newsKeywords)$ ;
9    $SIS \leftarrow calculateSentenceScore(WIS)$ ;
10   $keySentenceSet \leftarrow keySentencExtraction(SIS, T) \cup title$ ;
11   $KSW \leftarrow splitKeySentenceToWords(keySentenceSet)$ ;
12   $WOS \leftarrow calculateWordOrientationScore(Dict^A, KSW)$ ;
13   $SOS, numPosKS, numNegKS \leftarrow calculateSentenceOrientationScore(WOS)$ ;
14   $KValue, DValue, RSI \leftarrow calculateTechnicalIndicators(S, publishedDate)$ ;
15   $trendLabel \leftarrow generateStokTrend(S, P, publishedDate)$ ;
16   $generatedInstance \leftarrow mergeInformationByDate(SOS, KValue, DValue, RSI, trendLabel, publishedDate)$ ;
17  if  $numPosKS \geq numNegKS$  then
18     $posTrainDataset \leftarrow posTrainDataset \cup generatedInstance$ ;
19  else
20     $negTrainDataset \leftarrow negTrainDataset \cup generatedInstance$ ;
21 return  $posTrainDataset, negTrainDataset$ .
    
```

Fig. 1 shows that the proposed approach consists of three phases, namely data preprocessing, ensemble classifier creation, and prediction phase. In the data preprocessing phases, each Chinese news is first used to extract the K keywords and split them into a set of sentences. The similarity value of each sentence with the K keywords is calculated using word2vector. Based on the similarity values, the top T key phrases are selected. Using the sentiment dictionaries, the sentiment scores of the key phrases and the news title are calculated as classification attributes. In addition, the stock price dataset is used to calculate the technical indicators, including the K , D , and RSI values, as classification attributes, and the stock trend of a certain period as the class label. The generated $T + 1$ sentiment values, the K , D , and RSI values, and the stock trend designation are combined into a training instance based on the news release date and stored in the processed dataset. Based on the sentiment values of the news, they are split into two data sets. If the number of records with positive sentiment is greater than the number of records with negative sentiment, the

training instance is added to the positive news dataset. In contrast, it is assigned to the negative news dataset. In the phase where the ensemble classifier is created, the positive and negative classifiers are trained using the generated training datasets for positive and negative news. In addition, the 20-day moving average and stock series variance are used to construct the Bollinger bands as the third classifier. Finally, the three classifiers are used as an ensemble classifier to predict the stock trend by voting in the prediction phase. The following sections describe the details of the three phases.

B. Data Preprocessing

In the data preprocessing phase, which is the key phase of the proposed approach, the main objective is to generate positive and negative training instances through text analysis, technical indicators and sentiment orientation analysis based on the given Chinese news, stock prices and sentiment dictionaries to create the classifiers in the second phase. Following the earlier approach [31], the pseudocode for data preprocessing is shown in Algorithm 1.

TABLE I. A PROCESSED INSTANCE FOR THE PROPOSED APPROACH

SOS_T	SOS_1	SOS_2	SOS_3	SOS_4	SOS_5	SOS_6	SOS_7	K	D	RSI	$Trend$
2.69	3.15	1.74	2.96	1.94	2.91	3.11	3.09	60.27	51.88	85.71	1

Algorithm 1 shows that the preprocessing phase of the data consists of two parts, namely the extraction of key sentences (lines 4 to 10) and the analysis of sentiment in the news (lines 11 to 20). For the extraction of key phrases, the related information is generated from each news item, including the title, content, and publication date (line 4). The news *newskeywords* are then generated according to the specified parameter K and the content of the news (line 5). The content of each news item is split into sentences to create *sentenceSet* based on the predefined punctuation (line 6). Each sentence in *sentenceSet* is then converted into a set of words by jieba to obtain *wordSentenceSet* (line 7). The generated *newskeywords* and *wordSentenceSet* are used to calculate the word importance score WIS for each sentence in the *sentenceSet* (line 8). The sentence importance score SIS is calculated for each sentence using WIS (line 9). Key sentences are then selected based on SIS and the number of top key sentences T to obtain *keySentenceSet* (line 10).

For news sentiment analysis, each key sentence is divided into a set of keywords and stored in KWS (line 11) to calculate the sentiment value of each sentence. The prepared dictionaries and the KWS are used to compute the word orientation score WOS (line 12). The sentence orientation score SOS is then computed using the generated WOS (line 13), and the number of positive and negative key phrases $numPosKS$ and $numNegKS$ for each news is also recorded. To create a more reliable classifier, the technical indicators, including KD and RSI , are used as additional classification attributes and are calculated using the stock price series S and the published date of the news *publishedDate* (line 14). For each news, the trend label is generated based on the publication date of the stock price series S and the predefined stock price period P (line 15). Then the generated SOS , $KValue$, $DValue$, RSI , and *trendLabel* are merged according to the publication date *publishedDate* as a trading instance (line 16). If the number of positive key sets is greater than the number of negative key sets, the generated training instance is stored in *posTrain* record. Otherwise, it is stored in *negTrainDataset* (lines 17 to 20). Finally, the generated *posTrainDatasets* and *negTrainDatasets* are returned for the second phase (line 21). The details of the data preprocessing phase are described below.

1. Key Sentence Extraction

For the given Chinese news set $N = \{n_1, \dots, n_x\}$, the key phrase extraction for a news can be divided into the following steps. First, the title, content, and publication date are extracted from each news n_i and represented as title, content, and *publishedDate*. Then, *TextRank* is used to extract K keywords from the content of the news item, denoted as *newsKeywords* = $\{kw_1, kw_2, \dots, kw_k\}$. To find important sentences, the news content is split using punctuation marks such as $\{“”, “”, “”, “”\}$ into sentences to create a set of sentences *sentenceSet* = $\{sent_1, sent_2, \dots, sent_n\}$. Each sentence $sent_i$ in *sentenceSet* is used by Jieba to generate a set of words, noted as $sw_i = \{w_1, w_2, \dots, w_m\}$. In the same way, the word sets for all sentences are formed as *wordSentenceSet* = $\{sw_1, sw_2, \dots, sw_n\}$. For the $sent_i$ sentence, the similarity between it and the news content is calculated by the similarity of each word in sw_i and the *newsKeywords* using *word2vec* and denoted as *wordSentenceScore* $WIS_j = \{score_1^j, score_2^j, \dots, score_m^j\}$, where $score_m^j$ is the score of the f -th word in the $sent_i$ sentence to the keywords extracted from the news content and is calculated by the following equation 1:

$$score_f^j = \max \{sim(w_f, kw_1), sim(w_f, kw_2), \dots, sim(w_f, kw_k)\} \quad (1)$$

Then, the score of the sentence *sentScore_j* is calculated by average of the scores in the WIS_j using the equation 2:

$$sentScore_j = \frac{\sum_{f=1}^m score_f^j}{m} \quad (2)$$

In equation 2, the score of all sentences can be calculated and represented as $SIS = \{sentScore_1, sentScore_2, \dots, sentScore_n\}$. The top- T sentences based on the scores of sentences and the news title are selected as the key sentences to form the set *keySentenceSet* = $\{senTitle, kSent_1, \dots, kSent_j\}$ for news sentiment analysis.

2. News Sentiment Analysis

In this section, the details of news sentiment analysis are stated. For every news, the sentiment analysis can be processed by the following four steps, including the words extraction for key sentences, the word orientation score calculation for key sentences, the sentence orientation score calculation for key sentences, and the news orientation generation.

In the first step, each sentence $kSent_j$ in the *keySentenceSet* is split into a set of words and expressed as $kSent_j = \{ksw_{j1}, ksw_{j2}, \dots, ksw_{jl}\}$, where ksw_{ji} represents the i -th word of the key sentence, and L is the number of words. In the second step, using the given dictionaries $Dict^A$, which consist of the positive dictionary $Dict^P$ and the negative dictionary $Dict^N$, the word orientation score wos_{ji} of ksw_{ji} is calculated according to the following equation 3 as:

$$wos_{ji} = \tau \cdot MAX(SIM(ksw_{ji}, w^0)) \quad (3)$$

where $w^0 \in Dict^A$, and τ is set as 1 if $w^0 \notin Dict^P$; otherwise, it is set as -1 if $w^0 \in Dict^N$.

Based on the equation 3, this means that the wos_{ji} is set to the maximum similarity with the words in the dictionaries, and the *word2vec* is used to measure the similarity of two words in $SIM()$. If w^0 is included in $Dict^P$, then wos_{ji} is a negative value, otherwise it is a positive value. After calculating all the key phrases, this can be represented as $WOS = \{kSentWOS_1, \dots, kSentWOS_j, \dots, kSentWOS_{T+1}\}$, where $kSentWOS_j = \{wos_{j1}, \dots, wos_{j2}, \dots, wos_{jl}\}$. Meanwhile, the number of positive and negative sentences, $numPosKS$ and $numNegKS$, are also counted. In the third step, the sentence orientation value sos_j of a key sentence $kSent_j$ can be calculated using the following equation 4:

$$sos_j = \frac{\sum_{i=1}^L wos_{ji}}{L} \quad (4)$$

where L is the number of words of the key sentence $kSent_j$. Finally, the sentence orientation scores of all key sentences are calculated and denoted as $SOS = \{sos_1, \dots, sos_j, \dots, sos_{T+1}\}$. Then, according to the stock price series S and the published date of news *publishedDate*, the K , D , RSI values, and the stock trend label are generated. After that, the SOS , $KValue$, $DValue$, RSI , and *trendLabel* are merged according to the news published date as a trading instance. For example in Table I, the first eight columns are the sentiment scores of the key sentences and the last one is the class label.

In the news orientation generation, for every news, if the number of positive sentences is larger than the number of negative sentences, it is identified as a positive training instance. Otherwise, it is a negative training instance. In other words, according to the counted number of positive and negative sentences, the positive news training dataset *posNTrainDataset* and the negative news training dataset *negNTrainDataset* can be formed for next phase.

Algorithm 2: Ensemble classifier building, ClassifierBuildingProcedure()

Input: Stock prices dataset $S = \{s_1, \dots, s_y\}$, the positive news training dataset *posNTrainDataset*, the negative news training dataset *negNTrainDataset*, number of keywords K , number of top key sentences T , testind dataset proportion *testProportion*, and moving average days *maDay*.

Output: Stock trend ensemble classifier *sTrendEnsembleclassifier*.

```

1  posXTrain, posXTest, posYTrainLabel, posYTestLabel ← trainTestSplit(posNTrainDataset, testProportion);
2  negXTrain, negXTest, negYTrainLabel, negYTestLabel ← trainTestSplit(negNTrainDataset, testProportion);
3  accuracy ← [];
4  posNewClassifier ← buildClassifier(posXTrain, posYTrainLabel);
5  predictPosY ← newClassifierPrediction(posNewClassifier, posXTest);
6  accuracy ← classifierAccuracyScore(predictPosY, posYTestLabel);
7  negNewClassifier ← buildClassifier(negXTrain, negYTrainLabel);
8  predictNegY ← newClassifierPrediction(negNewClassifier, negXTest);
9  accuracy ← classifierAccuracyScore(predictNegY, posYTestLabel);
10 bbandClassifier ← generateBbandClassifier(S, maDay);
11 sTrendEnsembleclassifier ← posNewClassifier ∪ negNewClassifier ∪ bbandClassifier;
12 return sTrendEnsembleclassifier.
```

Algorithm 3: Prediction, PredictionProcedure()

Input: Stock prices dataset $S = \{s_1, \dots, s_y\}$, Chinese News $S = \{n_1, \dots, n_x\}$, sentiment dictionaries $Dict^A = \{dict^p, dict^n\}$, stock trend ensemble classifier *sTrendEnsembleclassifier*, number of keywords K , number of top key sentences T , and tradingThreshold α .

Output: Prediction $P = \{p_1, \dots, p_x\}$, $p \in \{0, 1\}$.

```

1  predictionResult ← [];
2  processedTestingDataset ← dataPreprocessing(S, N, DictA);
3  for data ∈ processedTestingDataset do
4      tradingSignal1 ← sTrendEnsembleclassifier.posClassifier(data);
5      tradingSignal2 ← sTrendEnsembleclassifier.negClassifier(data);
6      tradingSignal3 ← sTrendEnsembleclassifier.bbClassifier(S);
7      if tradingSignal1 + tradingSignal2 + tradingSignal3 ≥ α then
8          predictionResult.append(1); // Represent a buying signal
9      else
10         predictionResult.append(0); // Represent do nothing
11 return predictionResult.
```

C. Building Ensemble Classifier

In the proposed method, three classifiers are used to construct the ensemble classifier. The pseudo code of the second phase is shown in Algorithm 2.

From Algorithm 2, the ensemble classifier building procedure divides the preprocessed datasets, *posNTrainDataset* *negNTrainDataset*, into training and test datasets in the first phase according to the predefined test dataset proportion *testProportion* (lines 1 to 2). After the training and test datasets are created, the positive and negative classifiers are constructed using the Support Vector Machine (SVM) and their accuracies are stored in the accuracy list (lines 3 to 9). Then, the stock series S and the moving average days *maDay* are used to compute the Bollinger upper channel as the third classifier (line 10). Finally, the three classifiers are merged and the stock trend ensemble classifier *sTrendEnsembleclassifier* is returned (lines 11 to 12). In other words, the ensemble classifier is designed to take not only news contents and sentiments but also stock prices into consideration for generating more reliable trading signals for investors. For example, when a positive news is published, the prediction results of the first two classifiers based on the content of the news, K , D and RSI values can be used together to identify the trend of the target, which is better than the prediction by only one classifier. Hence, the first problem of the news-based forecasting model mentioned in the previous section can be solved. In addition, to make the trading signal more reliable, the technical indicator Bollinger bands is selected as the third classifier. If the stock price is greater than the upper Bollinger channel, a positive trading signal is generated. Otherwise, it generates a neutral trading signal.

D. Prediction Phase

After the stock trend ensemble classifier is built, it can be used for prediction. The pseudocode for the prediction phase is shown in Algorithm 3.

From Algorithm 3, in the prediction phase, when a news is released, using the same data preprocessing method, the key sentiment scores, $KValue$, $DValue$, and RSI are generated (line 2), and the processed data are used as input to the positive and negative classifiers to generate two trading signals, *tSignal1* and *tSignal2* (lines 4 to 5). Then, the stock price of the news release date is sent to the third classifier to generate the other trading signal, *tSignal3* (line 6). The final prediction is based on the voting of the three trading signals. If the voting score is greater than the trading threshold, a buy signal is generated. Because three classifiers are used in the ensemble classifier, when two out of three classifiers are suggesting buying, then the buying signal is outputted as the prediction result. On the contrary, when two of them suggesting selling, then selling signal is provided as the prediction result. Based on the signals, the return of every trading can be calculated.

IV. EXPERIMENTAL EVALUATION

In this section, we first describe the experimental data and settings in Section A. Then, in Sections B through D, the results of the proposed approach are examined with respect to different training intervals, different parameter settings, and in comparison to the existing approach.

A. Experimental Data and Settings

As for the financial corpus, we used real data for experiments to test the effectiveness of the proposed approach. The news information was obtained from one of the most popular financial websites covering the stock market in Taiwan. We selected relevant news about Catcher Technology Co, Ltd. and obtained 5,191 news articles from May 11, 2014 to January 11, 2020. However, the content of some news is not what we want. For example, the content of the news articles consists only of company-related data tables without any other text descriptions, or the length of the article is too short to extract a reasonable number of key sentences. So after we receive the news, we perform a screening action. The conditions for screening are as follows: (1) The length of the article must be more than 200 words. (2) The content must contain more than numerical data. After screening, there were 2,331 news articles left.

As for the stock price series, according to the time interval of the news, the time interval of our stock acquisition extends from May 1, 2014 to January 30, 2020, and the data was collected from the Taiwan Stock Exchange. The closing price of the stock used in this study is shown in Fig. 2.



Fig. 2. The closing price of the stock used in the experiments.

As for the sentiment dictionary, according to literature [39], the effect of using finance-related sentiment dictionaries in finance is better than that of general sentiment dictionaries. The sentiment dictionary used in this paper is a financial sentiment dictionary established by Loughran et al. [40], which includes a positive word dictionary and a negative word dictionary. Since the original text is in English, it needs to be translated and partially selected before use. The final positive dictionary and negative dictionary contain 352 and 2,332 words, respectively.

In the following experiments, we use standard evaluation indicators, including accuracy, recall, precision, and F-score. Accuracy can reflect the overall judgment of the model and can be calculated as follows by equation 5:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

where *true positive* (TP) refers to the number of data whose predicted trend is positive and the trend of those data is actually positive, *true negative* (TN) refers to the data whose trend is predicted

to be negative and the trend of those data is actually negative. *false positive* (FP) refers to the data whose predicted trend is positive but the actual trend is negative. *false negative* (FN) represents the data whose predicted trend is negative but the actual trend is positive. The precision, recall and F-score are respectively defined in equation 6, equation 7, and equation 8:

$$precision = \frac{TP}{TP + FP} \quad (6)$$

$$recall = \frac{TP}{TP + FN} \quad (7)$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (8)$$

where the *precision* is defined as the proportion of data that is recognized as a positive category. The recall rate is in all samples of positive categories, what is the proportion of correctly identified as positive categories. There is a close relationship between precision and recall. When the precision is low, the recall rate is usually higher; but when the precision is high, the recall rate is usually lower. Therefore, in practical applications, F-score is usually used to find a balance between the two. In addition, the profit-to-loss ratio (P/L ratio), which is calculated by the ratio of average profit to average loss, is defined in equation 9:

$$profit_loss = \frac{avg_profit}{avg_loss} \quad (9)$$

B. Evaluation on Different Training Intervals

Experiments with different lengths of training data to test whether the length of training time has an effect on prediction. Five different training periods are used in the experiments, including a 5-year training period (2014-2019), a 4-year training period (2015-2019), a 3-year training period (2016-2019), a 2-year training period (2017-2019), and a 1-year training period (2019). The test period is one year (2020). Table II shows the number of training and testing instances.

Moreover, the experiments are performed in a fixed way in terms of parameters. The number of keywords used to extract the key sentences is set to 10, and the number of key sentences T is set to 7 for testing. The prediction time is the same day, i.e., M is set to 0. In other words, the opening price minus the closing price is used to set the label. If the number is positive, it means an uptrend and is labeled "1", otherwise, it means a downward trend and is labeled as "0". The comparison results of the proposed approach with different number of classifiers for different training intervals are shown in Table III.

From Table III, we can make two observations: (1) Overall, the accuracy of the proposed approach using three classifiers is better than that using only one positive or negative classifier, especially when the 3-year training period is used to train the model. (2) From the profit-to-loss ratios, we can also see that the P/L ration of the proposed approach with a 5-year training period is 4.082, which is the highest compared to the others. Based on the results, we can say that the ensemble classifier derived with the proposed approach is effective. Note that the accuracies of the ensemble classifier, the negative model, and the positive model using 5-year dataset for training and 1-year dataset for testing are 0.6, 0.59, and 0.61, the results also indicate that there is no overfitting issue. Of course, when the number of training instances is too small, the overfitting problem may be occurred.

TABLE II. NUMBER OF TRAINING AND TESTING INSTANCES IN DIFFERENT INTERVALS

Training periods					Testing periods	Total
5 years	4 years	3 years	2 years	1 year	1 year	2 years
2014-2019	2015-2019	2016-2019	2017-2019	2019	2020	2014-2020
2,124	1,850	1,603	1,221	581	207	2,331

TABLE III. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT TRAINING INTERVALS

5 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
4 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.63	0.57	0.3	0.39	0.55	0.53	0.54	
0	0.68	0.43	0.53	0.56	0.8	0.66	0.61	0.63	0.62	
Accuracy	0.59			0.56			0.58			3.4097
3 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.51	0.8	0.62	0.57	0.3	0.39	0.57	0.55	0.56	
0	0.67	0.34	0.45	0.56	0.8	0.66	0.62	0.64	0.63	
Accuracy	0.55			0.56			0.6			3.6884
2 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.51	0.79	0.62	0.56	0.38	0.45	0.54	0.56	0.55	
0	0.66	0.35	0.46	0.57	0.73	0.64	0.6	0.58	0.59	
Accuracy	0.55			0.56			0.57			2.9368
1 year	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.48	0.86	0.62	0.55	0.4	0.46	0.53	0.53	0.53	
0	0.64	0.22	0.32	0.57	0.71	0.63	0.59	0.59	0.59	
Accuracy	0.51			0.56			0.56			2.5783

TABLE IV. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT NUMBER OF KEYWORDS

		Training data					Testing data
		5 years	4 years	3 years	2 years	1 year	1 year
K = 10	Total	2125	1851	1604	1222	582	206
	Positive	1345	1164	1020	811	413	135
	Negative	780	687	584	411	169	71
K = 8	Positive	1318	1148	1004	786	403	137
	Negative	807	703	600	436	179	69
K = 6	Positive	1343	1163	1015	803	403	137
	Negative	782	688	589	419	179	69
K = 4	Positive	1343	1163	1015	803	403	137
	Negative	782	688	589	419	179	69

C. Evaluation on Different Parameter Settings

In this section, the proposed approach is tested with different parameter settings. The parameters include the number of keywords K extracted from the key sentence K , the number of key sentences T , the prediction time point M , and the standard deviation γ for Bollinger bands. Therefore, a total of four experiments were conducted in this subsection.

First, experiments were conducted to show the influence of the number of keywords on the proposed approach. Other parameters such as the number of key phrases T , prediction time M , training interval Y , and standard deviation γ are set to 7, 0, 5, and 1, respectively. When the number of keywords K is different, the results for the number of positive and negative news will also different. Table IV shows that the number of positive and negative news can be obtained with different number of keywords from 4 to 10.

Table IV shows that when the number of keywords is less than 6, the number of positive and negative news generated is the same, indicating that there are too few keywords to perform effective sentiment analysis, so that a state of convergence occurs. The

experimental results of the proposed approach with different number of keywords are shown in Table V.

From Table V, we can see that the best P/L ratio is 4.0821 when the number of keywords for the derived ensemble classifier was set to 10 by the proposed approach. The results also show that the number of keywords indeed has an impact on the construction of the classifier.

Experiments were then conducted to show the influence of the number of key sentences on the proposed approach. Other parameters, including the number of words K , prediction period M , training interval Y , and standard deviation γ are set to 10, 0, 5, and 1, respectively. The experimental results of the proposed approach with different number of key sentences from 4 to 7 are shown in Table VI.

Table VI shows that the P/L ratio and accuracy of the proposed approach are highest when the number of key sentences is set to 7. When the number of key sentence were set to 4 to 6, it can be seen that the results seem to be somewhat unstable in terms of P/L ratio.

Next, experiments were conducted to show the influence of prediction periods on the proposed approach. Other parameters such as the number of words K , the number of key sentences T , the training

TABLE V. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT NUMBER OF KEYWORDS

K = 10	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
K = 8	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.58	0.82	0.68	0.6	0.43	0.5	0.57	0.6	0.59	
0	0.78	0.53	0.63	0.55	0.71	0.62	0.64	0.61	0.62	
Accuracy	0.66			0.57			0.61			3.2891
K = 6	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.8	0.66	0.64	0.39	0.480.56	0.55	0.56		
0	0.77	0.52	0.62	0.53	0.76	0.62	0.62	0.63	0.62	
Accuracy	0.64			0.57			0.59			3.9313
K = 4	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.8	0.66	0.64	0.39	0.48	0.56	0.55	0.56	
0	0.77	0.52	0.62	0.53	0.76	0.62	0.62	0.63	0.62	
Accuracy	0.64			0.57			0.59			3.9313

TABLE VI. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT NUMBER OF KEY SENTENCES

T = 7	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
T = 6	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.55	0.75	0.64	0.58	0.38	0.45	0.58	0.57	0.58	
0	0.69	0.48	0.56	0.58	0.76	0.65	0.63	0.64	0.63	
Accuracy	0.6			0.58			0.61			3.6
T = 5	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.55	0.75	0.64	0.58	0.28	0.37	0.57	0.53	0.55	
0	0.69	0.48	0.56	0.56	0.82	0.67	0.61	0.65	0.63	
Accuracy	0.6			0.56			0.59			3.8661
T = 4	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.77	0.65	0.54	0.33	0.41	0.58	0.51	0.54	
0	0.7	0.48	0.57	0.56	0.76	0.64	0.61	0.67	0.64	
Accuracy	0.61			0.55			0.6			3.8225

TABLE VII. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT NUMBER OF KEY SENTENCES

M = 0	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
M = 3	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.32	0.14	0.2	0.64	0.64	0.64	0.52	0.37	0.43	
0	0.5	0.74	0.6	0.7	0.7	0.7	0.5	0.66	0.57	
Accuracy	0.46			0.67			0.51			1.0247
M = 5	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.52	0.41	0.46	0.54	0.88	0.67	0.58	0.63	0.6	
0	0.57	0.68	0.62	0.71	0.28	0.4	0.56	0.52	0.54	
Accuracy	0.55			0.53			0.57			1.5192

TABLE VIII. COMPARISON RESULTS OF THE PROPOSED APPROACH IN TERMS OF VARIOUS CRITERIA USING DIFFERENT γ VALUES

$\gamma = 1$	BBAND			P/L Ratio	PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	0.6	0.52	0.56		0.58	0.61	0.6	
0	0.63	0.7	0.66		0.66	0.61	0.63	
Accuracy	0.62			3.2681	0.61			4.0821
$\gamma = 2$	BBAND			P/L Ratio	PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	0.84	0.32	0.47		0.58	0.54	0.56	
0	0.62	0.95	0.75		0.62	0.66	0.64	
Accuracy	0.66			4.2631	0.61			4.2833
$\gamma = 3$	BBAND			P/L Ratio	PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	1	0.49	0.66		0.58	0.53	0.55	
0	0.54	1	0.7		0.62	0.66	0.64	
Accuracy	0.55			N/A	0.6			4.1916

TABLE IX. COMPARISON RESULTS OF THE PROPOSED APPROACH AND GASTP IN TERMS OF VARIOUS CRITERIA USING DIFFERENT M VALUES

$M = 0$	Proposed Approach			P/L Ratio	GASTP			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	0.58	0.62	0.6		0.58	0.42	0.49	
0	0.65	0.6	0.62		0.55	0.7	0.61	
Accuracy	0.61			4.0821	0.56			3.725
$M = 3$	Proposed Approach			P/L Ratio	GASTP			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	0.52	0.37	0.43		0.64	0.2	0.31	
0	0.5	0.66	0.57		0.51	0.88	0.65	
Accuracy	0.54			1.0247	0.53			2.6666
$M = 5$	Proposed Approach			P/L Ratio	GASTP			P/L Ratio
	Precision	Recall	F1		Precision	Recall	F1	
1	0.58	0.63	0.6		0.6	0.53	0.56	
0	0.56	0.52	0.54		0.54	0.61	0.57	
Accuracy	0.57			1.5192	0.57			1.2921

interval Y , and the standard deviation γ were set to 10, 7, 5, and 1, respectively. Table VII shows the experimental results of the algorithm at different prediction periods, which are 0, 3, and 5.

From Table VII, we can easily see that the P/L ratio of the proposed approach at a prediction period of 0 is 4.0821, which is the highest and better than the other two settings. Moreover, although the accuracy of the negative model is the highest at a prediction period of 3, the P/L ratio is the worst. From these experimental results, we can conclude that the effect of predicting the rise and fall of the day is the best. In other words, the financial news is suitable for predicting short-term stock trends.

Finally, experiments were conducted to show the influence of the weight of the standard deviation of the Bollinger bands on the proposed approach. Other parameters, such as the number of words K , the number of key sentences T , the prediction time period M , and the training interval Y were set to 10, 7, 0, and 5, respectively. Table VIII shows the experimental results of the proposed algorithm at different γ -values by 0, 3, and 5.

Table VIII shows that the proposed approach can achieve the best values for the P/L ratio when γ was set to 2. When γ was set to 3, the generated trading signal from the Bollinger Bands is "do not buy" and consequently the P/L ratio is zero. However, with the derived ensemble classifier, an acceptable P/L ratio can still be achieved, which means that the ensemble classifier is better than using a single classifier.

D. Comparisons With the State-of-the-art Models

In this section, experiments were conducted to show the results of the comparison between the proposed approach which is a sentence-level approach and the previous approach [16], where a keyword-based method, technical indicators and genetic algorithms were considered together to adjust the parameters for generating trading signals, called GASPT. The results of the comparison are shown in Table IX.

Table IX shows that the proposed approach is better than GASTP in terms of P/L ratio when the prediction period was set to 0. However, when M was set to 3, GASTP is better than the proposed approach. Based on the experimental results, we can conclude that the proposed approach is suitable and effective for short-term trading. Besides, in the proposed approach, the time information is important for generating training datasets, including the classification attributes and labels. It may also be utilized to reduce the training dataset when large amount of news are given. In other words, it is worthy to design a novel algorithm to increase the efficiency of the proposed approach in the future.

V. CONCLUSION AND FUTURE WORK

This paper proposes a three-stage framework, including data preprocessing, ensemble classifier creation, and prediction phases, for predicting stock trends based on sentence-level sentiment analysis of Chinese news and technical indicators. In first phase, the goal is to generate the training instances from given corpus and stock

prices. Then, based on the sentiment orientation of instances, the positive and negative classifiers are constructed. To increase the reliability of the prediction result, Bollinger bands are also considered as another classifier. The three classifiers are merged as a final ensemble classifier to generate trading signals. Experiments were also conducted on the real data set. The experimental results show that: (1) The longer the training period, the higher the accuracy and profit-to-loss ratio; (2) The number of keywords used to find the key sentences should be larger than a threshold value for the classifier to achieve more stable results; (3) The constructed ensemble classifier is suitable for predicting the stock trend on the news release day; (4) Compared with the previous key-level based prediction approach, the proposed sentence-level based approach is effective in terms of accuracy and the P/L ratio. In the future, the proposed approach can be further developed in the following directions. For example, the optimization algorithm can be applied to it to automatically find a suitable parameter setting, transfer learning can also be used to select useful instances from the given sources to improve the accuracy of the classifier, or weight mechanism can be designed for the positive and negative models to derive a better performance.

REFERENCES

- [1] M. Drenovak, V. Ranković, B. Urošević, R. Jelic, "Mean- maximum drawdown optimization of buy-and-hold portfolios using a multi-objective evolutionary algorithm," *Finance Research Letters*, p. 102328, 2021.
- [2] T. E. Simos, S. D. Mourtas, V. N. Katsikis, "Time- varying black-litterman portfolio optimization using a bio-inspired approach and neurons," *Applied Soft Computing*, p. 107767, 2021.
- [3] S. T. Tayah, "A novel backtesting methodology for clustering in mean-variance portfolio optimization," *Knowledge-Based Systems*, vol. 209, p. 106454, 2021.
- [4] W. Ding, K. Mazouz, Q. Wang, "Volatility timing, sentiment, and the short-term profitability of vix-based cross-sectional trading strategies," *Journal of Empirical Finance*, vol. 63, pp. 42–56, 2021.
- [5] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, H. Fujita, "Adaptive stock trading strategies with deep reinforcement learning methods," *Information Sciences*, vol. 538, pp. 142–158, 2020.
- [6] A. K. Banerjee, A. Dionisio, H. K. Pradhan, B. Mahapatra, "Hunting the quicksilver: Using textual news and causality analysis to predict market volatility," *International Review of Financial Analysis*, p. 101848, 2021.
- [7] N. Pröllochs, S. Feuerriegel, D. Neumann, "Negation scope detection in sentiment analysis: Decision support for news-driven trading," *Decision Support Systems*, vol. 88, pp. 67–75, 2016.
- [8] W. Chen, M. Jiang, W. G. Zhang, Z. Chen, "A novel graph convolutional feature based convolutional neural network for stock trend prediction," *Information Sciences*, vol. 556, pp. 67–94, 2021.
- [9] J. Long, Z. Chen, W. He, T. Wu, J. Ren, "An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in chinese stock exchange market," *Applied Soft Computing*, vol. 91, p. 106205, 2020.
- [10] X. Zhang, A. Li, R. Pan, "Stock trend prediction based on a new status box method and adaboost probabilistic support vector machine," *Applied Soft Computing*, vol. 49, pp. 385–398, 2016.
- [11] X. Li, H. Xie, L. Chen, J. Wang, X. Deng, "News impact on stock price return via sentiment analysis," *Knowledge-Based Systems*, vol. 69, pp. 14–23, 2014.
- [12] G. Löffler, L. Norden, A. Rieber, "Negative news and the stock market impact of tone in rating reports," *Journal of Banking & Finance*, vol. 133, p. 106256, 2021.
- [13] S. Kelly, K. Ahmad, "Estimating the impact of domain-specific news sentiment on financial assets," *Knowledge-Based Systems*, vol. 150, pp. 116–126, 2018.
- [14] A. Thakkar, K. Chaudhari, "Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks," *Applied Soft Computing*, vol. 96, p. 106684, 2020.
- [15] L. Lei, "Wavelet neural network prediction method of stock price trend based on rough set attribute reduction," *Applied Soft Computing*, vol. 62, pp. 923–932, 2018.
- [16] C. H. Chen, P. Shih, G. Srivastava, S. T. Hung, J. C. W. Lin, "Evolutionary trading signal prediction model optimization based on chinese news and technical indicators in the internet of things," *IEEE Internet of Things Journal*, p. Early access, 2021.
- [17] G. Liu, X. Wang, "A new metric for individual stock trend prediction," *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 1–12, 2019.
- [18] P. Y. Hao, C. F. Kung, C. Y. Chang, J. B. Ou, "Predicting stock price trends based on financial news articles and using a novel twin support vector machine with fuzzy hyperplane," *Applied Soft Computing*, vol. 98, p. 106806, 2021.
- [19] W. Long, L. Song, Y. Tian, "A new graphic kernel method of stock price trend prediction based on financial news semantic and structural similarity," *Expert Systems with Applications*, vol. 118, pp. 411–424, 2019.
- [20] P. Rajendiran, P. L. K. Priyadarsini, "Survival study on stock market prediction techniques using sentimental analysis," *Materials Today: Proceedings*, p. Early access, 2021.
- [21] C. Martin, D. Quintana, P. Isasi, "Dynamic generation of investment recommendations using grammatical evolution," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, pp. 104–111, 2021.
- [22] L. Chen, L. Sun, C. M. Chen, M. E. Wu, J. M. T. Wu, "Stock trading system based on machine learning and kelly criterion in internet of things," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–9, 2021.
- [23] J. M. T. Wu, L. Sun, G. Srivastava, J. C. W. Lin, "A long short-term memory network stock price prediction with leading indicators," *Big Data*, vol. 9, pp. 343–357, 2021.
- [24] J. M. T. Wu, S. Sun, L. G., J. C. W. Lin, "A novel synergetic lstm-ga stock trading suggestion system in internet of things," *Mobile Information Systems*, vol. 2021, pp. 1–15, 2021.
- [25] H. Huang, X. Liu, Y. Zhang, F. C., "News-driven stock prediction via noisy equity state representation," *Neurocomputing*, vol. 470, pp. 66–75, 2022.
- [26] A. Derakhshan, H. Beigy, "Sentiment analysis on stock social media for stock price movement prediction," *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 569–578, 2019.
- [27] Y. Li, H. Bu, J. Li, J. Wu, "The role of text-extracted investor sentiment in chinese stock price prediction with the enhancement of deep learning," *International Journal of Forecasting*, vol. 36, pp. 1541–1562, 2020.
- [28] C. Song, X. Wang, P. Cheng, J. Wang, L. Li, "Sapcc: A framework based on probabilistic linguistic terms for short text sentiment analysis," *Knowledge-Based Systems*, vol. 194, p. 105572, 2020.
- [29] J. Wu, K. Lu, S. Su, S. Wang, "Chinese micro- blog sentiment analysis based on multiple sentiment dictionaries and semantic rule sets," *IEEE Access*, vol. 7, pp. 183924–183939, 2019.
- [30] W. Wang, D. Huang, J. Cao, "Chinese syntax parsing based on sliding match of semantic string," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 7, pp. 1–14, 2020.
- [31] P. Wang, Y. Luo, Z. Chen, L. He, Z. Zhang, "Orientation analysis for chinese news based on word embedding and syntax rules," *IEEE Access*, vol. 7, pp. 159888–15898, 2019.
- [32] S. Wang, J. Zhang, C. Zong, "Empirical exploring word-character relationship for chinese sentence representation," *ACM Transactions on Asian and Low- Resource Language Information Processing*, vol. 17, pp. 1–18, 2019.
- [33] F. Chen, Y. Huang, "Knowledge-enhanced neural networks for sentiment analysis of chinese reviews," *Neurocomputing*, vol. 368, pp. 51–58, 2019.
- [34] T. Daudert, "Exploiting textual and relationship information for fine-grained financial sentiment analysis," *Knowledge-Based Systems*, p. 107389, 2021.
- [35] G. Xu, Y. Meng, X. Qiu, Z. Yu, X. Wu, "Sentiment analysis of comment texts based on bilstm," *IEEE Access*, vol. 7, pp. 51522–51532, 2019.
- [36] J. Yang, R. Yang, H. Lu, C. Wang, J. Xie, "Multi- entity aspect-based sentiment analysis with context, entity, aspect memory and dependency information," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, pp. 1–22, 2019.
- [37] M. Pota, H. Ventura, M. amd Fujita, M. Esposito, "Multilingual evaluation of pre-processing for bert- based sentiment analysis of tweets," *Expert Systems with Applications*, vol. 181, p. 115119, 2021.
- [38] H. Peng, Y. Ma, S. Poria, Y. Li, E. Cambria, "Phonetic- enriched text representation for chinese sentiment analysis with reinforcement learning," *Information Fusion*, vol. 70, pp. 88–99, 2021.

- [39] A. Li, P. Wu, W. Wang, "Incorporating stock prices and news sentiments for stock market prediction: A case of hong kong," *Information Processing & Management*, vol. 57, p. 102212, 2020.
- [40] T. Loughran, B. McDonald, "When is a liability not a liability? textual analysis, dictionaries, and 10-ks," *Cognitive Computation*, pp. 1167-1176, 2011.

Chun-Hao Chen



Chun-Hao Chen is an associate professor at Department of Information and Finance Management at National Taipei University of Technology, Taipei, Taiwan. Dr. Chen received his Ph.D. degree with major in computer science and information engineering from National Cheng Kung University, Taiwan, in 2008. He has a wide variety of research interests covering data mining, time series, machine learning, evolutionary algorithms, and fuzzy theory. Research topics cover portfolio selection, trading strategy, business data analysis, time series pattern discovery, etc. He serves as the associate editor of the International Journal of Data Science and Pattern Recognition, and IEEE Access. He is also a member of IEEE.

Po-Yeh Chen



Po-Yeh Chen received her M.S. degree in Computer Science from the Department of Computer Science and Information Engineering at Tamkang University, Taiwan, in 2020. He now is a technical engineer of Comwave International Ltd. His research interests include text mining and classification with a focus on financial sector.

Jerry Chun-Wei Lin



Jerry Chun-Wei Lin received his Ph.D. from the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan in 2010. He is currently a full Professor with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway. He has published more than 500+ research articles in refereed journals (IEEE TKDE, IEEE TFS, IEEE TNNLS, IEEE TCYB, IEEE TII, IEEE TITS, IEEE TNSE, IEEE TETCI, IEEE SysJ, IEEE SensJ, IEEE IOTJ, ACM TKDD, ACM TDS, ACM TMIS, ACM TOIT, ACM TIST, ACM TOSN) and international conferences (IEEE ICDE, IEEE ICDM, PKDD, PAKDD), 15 edited books, as well as 33 patents (held and filed, 3 US patents). His research interests include data mining and analytics, natural language processing (NLP), soft computing, IoTs, bioinformatics, artificial intelligence/machine learning, and privacy preserving and security technologies. He is the Editor-in-Chief of the International Journal of Data Science and Pattern Recognition, the Guest Editor/Associate Editor for several IEEE/ACM journals such as IEEE TFS, IEEE TII, IEEE TIST, IEEE JBHI, ACM TMIS, ACM TOIT, ACM TALLIP, and ACM JDIQ. He has recognized as the most cited Chinese Researcher respectively in 2018, 2019, and 2020 by Scopus/Elsevier. He is the Fellow of IET (FIET), ACM Distinguished Member (Scientist), and IEEE Senior Member.