Check for
updates

A blending ensemble learning model for crude oil price forecasting

Mahmudul Hasan[1] · Mohammad Zoynul Abedin[2] · Petr Hajek[3] ·
Kristof Coussement[4] · Md. Nahid Sultan[1] · Brian Lucey[5]

Abstract

To efficiently capture diverse fluctuation profiles in forecasting crude oil prices, we here propose to combine heterogenous predictors for forecasting the prices of crude oil. Specifically, a forecasting model is developed using blended ensemble learning that combines various machine learning methods, including k-nearest neighbor regression, regression trees, linear regression, ridge regression, and support vector regression. Data for Brent and WTI crude oil prices at various time series frequencies are used to validate the proposed blending ensemble learning approach. To show the validity of the proposed model, its performance is further benchmarked against existing individual and ensemble learning methods used for predicting crude oil price, such as lasso regression, bagging lasso regression, boosting, random forest, and support vector regression. We demonstrate that our proposed blending-based model

✉ Mohammad Zoynul Abedin
    m.z.abedin@swansea.ac.uk

    Mahmudul Hasan
    mahmudulmoon123@gmail.com

    Petr Hajek
    petr.hajek@upce.cz

    Kristof Coussement
    k.coussement@ieseg.fr

    Md. Nahid Sultan
    nahid.sultan@hstu.ac.bd

    Brian Lucey
    brian.lucey@tcd.ie

[1] Department of Computer Science and Engineering, Hajee Mohammad Danesh Science and
     Technology University, Dinajpur 5200, Bangladesh

[2] Department of Accounting and Finance, School of Management, Swansea University, Fabian Way,
     Bay CampusSwansea, Wales SA1 8EN, UK

[3] Science and Research Centre, Faculty of Economics and Administration, University of Pardubice,
     532 10 Pardubice, Czech Republic

[4] IESEG School of Management, Univ. Lille, CNRS, UMR 9221 - LEM - Lille Economie
     Management, Lille F-59000, France

[5] Trinity Business School, Trinity College Dublin, Dublin 2, Ireland

dominates the existing forecasting models in terms of forecasting errors for both short- and medium-term horizons.

1 Introduction

Crude oil is essentially the primary source of today's major oils and fuels. It is a petroleum product that consists of organic components and natural hydrocarbon reserves. Crude oil is found in subsurface reservoirs as a liquid. It is considered the primary fuel of the world, contributing almost one-third of global consumption of energy. An increase in crude oil prices may exert downward pressure on projected economic growth rates, potentially contributing to inflationary pressures. While a downturn in oil prices may negatively affect the economic development of oil-exporting countries, an increase can pose challenges for the economic growth of oil-importing nations (Behmiri & Manso, 2013). These fluctuations can subsequently influence economic growth forecasts. Altered growth projections may then impact expectations of corporate profitability, which could, in turn, exert influence on oil price dynamics. Moreover, volatility in crude oil prices has a significant impact on other economic activities, as crude oil is the main source of energy (Zhao et al., 2017). As a result, crude oil greatly impacts the world economy and stability (Chen & Huang, 2021; Gao et al., 2017). The price of crude oil has also proven to be a valuable indicator for assessing the relative predictive performance of stock markets (Nonejad, 2021). Therefore, predicting crude oil prices is an urgent task for researchers as well as companies, industries, and governments.

Different types of approaches have been developed for oil price forecasting over the past decades. Statistical approaches, econometric approaches, machine learning, deep learning, and ensemble learning approaches have been utilized to model the intrinsic complexity of the oil prices. Most of the studies used neural networks (NN) (Boubaker et al., 2022; Cerqueti & Fanelli, 2021; Moshiri & Foroutan, 2006) and support vector machines (SVM) (Zha et al., 2021; Ibrahim et al., 2022) to predict the oil price. Evolutionary-based learning (Mostafa & El-Masry, 2016; Sun et al., 2022) and the combination of supervised and unsupervised learning (Shin et al., 2013) have also been used to estimate oil prices. Particular attention is increasingly being paid to ensemble learning, a paradigm in which the resulting forecasting combines the outputs of multiple heterogeneous or homogeneous underlying predictors. Bootstrap aggregation (bagging) of stacked denoising autoencoders is an example of homogeneous ensemble models (Zhao et al., 2017). The heterogeneous ensemble of exponential smoothing and multilayer perceptron neural network was also used for forecasting crude oil price (Sun et al., 2018).

However, none of the above methods was dominant in terms of forecasting accuracy for longer time periods, due to the high diversity in price fluctuations (Gao et al., 2017; Li et al., 2021). High volatility in crude oil prices is mainly due to fluctuations in supply and demand, political instability, actions by organizations such as OPEC, currency fluctuations, speculative trading and market sentiment (Chang et al., 2022). Even in comparison with other commodity prices, oil price fluctuations are more susceptible to economic uncertainty (Bakas & Triantafyllou, 2020). Moreover, the volatility of crude oil prices exhibits significant structural shifts over time. This volatility is influenced by both short-term and

long-term informational factors (Wen et al., 2016). Therefore, to achieve high levels of forecasting accuracy, it is imperative to employ an integrative modeling approach that synthesizes predictions from a diverse set of sub-models. Although previous studies have attempted to address this problem using ensemble learning approaches, homogeneous sets of predictors have prevailed (Yu et al., 2016; Zhu et al., 2017), which are not effective for modelling different profiles of crude oil price behavior. To overcome this problem, here we propose using blending ensemble learning, a computationally efficient variant of stacking regression. The blending approach applies a leave-out method for individual weak predictors to combine the strengths of multiple diverse models, which results in better forecasting results than individual machine learning models. Hence, blending ensemble learning has received much attention over the last years in various application domains, including agriculture (Wu et al., 2021a, 2021b), pattern recognition (Hao et al., 2020), and stock market prediction (Li & Pan, 2022). Despite this interest in forecasting crude oil prices, no one, to the best of our knowledge, has investigated the performance of blending ensemble learning in this domain.

To bridge the gap in the literature, here we propose a blending ensemble machine learning model that combines five diverse predictors, namely, k-nearest neighbor regression, linear regression, regression tree, support vector regression, and ridge regression. Hereafter, we will refer to the new blending ensemble model as LKDSR by the initial letters of the methods used.

In this study, we analyze the two most commonly used crude oil data of Brent and West Texas Intermediate (WTI) for forecasting. Daily data are converted into weekly and monthly models for short- and medium-term forecasting. Statistical characteristics of the Brent and WTI time series data are analyzed, and necessary data preprocessing and mode changes are performed to enable predictions at different time series periods. The main contributions of this paper are listed below:

1. A multiscale model based on blending ensemble learning is proposed that predicts the short-term and medium-term crude oil prices, which allows us to break the limitations of a single time series decomposition analysis.
2. The superiority of the proposed blending ensemble model is demonstrated compared to the state-of-the-art crude oil price forecasting models used as a reference. The results of the Diebold Mariano (DM) test confirm the dominance of the proposed forecasting model.

The remaining parts of this manuscript are organized as follows. Section 2 presents the literature review of related works. Section 3 provides a description of the data used and their pre-processing. Section 4 outlines the methods used. Section 5 presents the results and analysis of empirical experiments. In Sect. 6, we discuss the results obtained, and Sect. 7 presents the conclusions and suggests future research avenues.

2 Related literature

In recent years, there has been remarkable work on crude oil price forecasting. Many researchers have shown promising results based on different types of oil price time series analysis using statistical models (Herrera et al., 2018; Rubaszek, 2021), econometrics models (Asai et al., 2020), and machine learning models (Zhang et al., 2022). In this section, we focus on recent advances in crude oil price forecasting using machine learning methods, including deep learning and hybrid and ensemble models. A comparison of the performance of these methods in predicting the price of crude oil is presented in Table 1.

Table 1 Performance comparison of existing machine learning-based crude oil forecasting models

Study	Data	Model	Results
Ghaffari and Zare (2009)	WTI	ANFIS	Acc = 68.18%
Haruna et al. (2015)	WTI	GA-NN	MSE = 0.0000001, RMSE = 0.0012
Lu et al. (2021)	WTI	Variable selection-LSTM	MAPE = 0.74, RMSE = 1.12
Chen et al. (2017)	WTI	RW-DBN	MSE = 0.0005
Zhao et al. (2017)	WTI	SDAE-Bagging	MAPE = 0.053, RMSE = 4.9
Huang and Wang (2018)	Brent, WTI	RWNN	RMSE = 2.05, 2.44, MAPE = 2.56, 1.81, sMAPE = 2.49, 2.99
Cen and Wang (2019)	Brent, WTI	LSTM	RMSE = 0.45, 0.38, MAE = 0.18, 0.17, MAPE = 3.46, 3.71, sMAPE = 3.86, 3.86
Norouzi and Fani (2020)	OPEC price	Gray model	MAE = 0.21, MSE = 0.089
Vo et al. (2020)	Brent	Bi-LSTM	MSE = 2.4, RMSE = 1.55, MAE = 1.2, MAPE = 2.50
Bristone et al. (2020)	Brent, WTI	Complex network deep learning	RMSE = 2.18, 2.08
Wang et al. (2020)	Brent	LSTM (Bi-LSTM) + GWO	PICP = 98.90%
Wu et al., (2021a, 2021b)	WTI	CNN	MAPE = 0.057
Zhao et al. (2021)	WTI, Brent, INE	PVMD-SVM-ARMA	MAE = 0.002, 0.001, 0.001, MSE = 0.011, 0.010, 0.004
Yang et al. (2021)	WTI	k-means + KPCA + KELM	RMSE = 3.79, MAPE = 0.0021
Busari and Lim (2021)	Export spot price	AdaBoost-LSTM, AdaBoost-GRU	MAE = 1.42, RMSE = 2.46, MAPE = 0.35

Acc accuracy, *ANFIS* adaptive network-based fuzzy inference system, *CEEMDAN* complementary ensemble empirical mode decomposition with adaptive noise, *GWO* grey wolf optimizer, *INE* international energy exchange, *PICP* forecasting interval coverage probability, *RWNN* random wavelet neural network

2.1 Shallow neural network models

By combining the learning capacity of NNs and the interpretability of fuzzy rule-based systems, Ghaffari and Zare (2009) proposed a unique soft computing-based approach to predict fluctuations in the WTI crude oil price. However, such a model has limited forecasting accuracy, as it suffers from the limitations of shallow neural networks. To avoid the problems associated with the use of the backpropagation algorithm to train shallow neural networks, Chen (2022) showed that much better computational efficiency can be achieved by using evolutionary algorithms to train NN forecasting models. Furthermore, Huang and Wang (2018) combined random wavelet NNs with a random time effective function that effectively

exploits historical crude oil time series data. Their empirical results confirmed the advantages of the proposed model over traditional shallow NNs and SVMs. To overcome the problems of slow convergence and parameter sensitivity of conventional feed-forward NNs, Bisoi et al. (2019) proposed a random vector functional link network (RVFLN) non-iterative approach for crude oil price forecasting. However, RVFLN models are susceptible to overfitting, and the training process of RVFLN is still subject to reaching local minima.

2.2 Deep learning models

To capture complex and rich features from crude oil data, deep learning-based forecasting models were proposed, such as those using bidirectional long short-term memory (Bi-LSTM) (Cen & Wang, 2019; Vo et al., 2020), which, unlike LSTM (Karasu & Altan, 2022; Lu et al., 2021), exploit the time series information in both directions, forward and backward. A hybrid model of LSTM and complex network analysis was introduced to map and reconstruct crude oil price data, resulting in an increase in accuracy of up to 58.3% (Bristone et al., 2020).

However, it is worth noting that LSTM and Bi-LSTM are computationally intensive and slow models whose forecasting performance was surpassed by ensembles of neural networks (Abedin et al., 2021; Busari & Lim, 2021). Specifically, Busari and Lim (2021) compared two homogeneous ensemble models, AdaBoost-LSTM and AdaBoost-GRU, and the empirical results suggest that AdaBoost-GRU outperforms AdaBoost-LSTM in predicting crude oil prices. Wang et al. (2020) proposed the improved grey wolf optimizer to integrate the forecasting intervals obtained using LSTM and Bi-LSTM into a reliable interval that captures the uncertainty present in the price of crude oil.

2.3 Hybrid machine learning models

Zhao et al. (2021) developed a new hybrid approach that can enable online real-time fore-casting of crude oil prices. They concluded that the oil market is so unpredictable that it resembles an imbalance of time series data. They proposed a PVMD-SVM-ARIMA model by combining particle swarm optimization, VMD parameter optimizer, SVM and ARIMA statistical model. It is suitable for short-term oil price prediction, but it is weak in handling the large data. Among some fusion models, it shows lower MSE and high accuracy in prediction. Yang et al. (2021) found that the use of divide-and-conquer technique improves the accuracy of the prediction. To this end, they developed a hybrid technique combining extreme learning machine, principal component analysis, and k-means. In this hybrid stream of research, Chen et al. (2017) used a hybrid model that combined the traditional ARMA (autoregressive moving average) model with LSTM and deep belief networks to capture both the linear and non-linear patterns in the crude oil time series data. To further improve prediction accuracy, additional sources of data were suggested. To illustrate, Li et al. (2019) and Wu et al., (2021a, 2021b) proposed revolutionary big-data-driven and text-based techniques that used a convolutional neural network (CNN) to automatically scrap crude oil news updates. To construct a text-based crude oil forecasting system, more than eight thousand news headlines were collected by Wu et al., (2021a, 2021b). Furthermore, to incorporate the Internet investor concern over the crude oil market, Google trend search data were utilized by Wang et al. (2018).

Ensemble models have recently shown striking results in forecasting crude oil prices (Escribano & Wang, 2021). Yu et al. (2016) proposed an ensemble of empirical mode decompositions that uses extended extreme learning machine (ELM) to divide continuous time series

data from crude oil into discrete regular data. The combination of extended ELM and ensemble empirical mode decomposition was shown to save time and to be robust to highly volatile time series data. Zhao et al. (2017) combined the stacked denoising autoencoder (SDAE) and bootstrap aggregation (bagging) ensemble method to predict the WTI crude oil price. They showed that their approach outperforms some competing methods, including the bagging of NNs and SVRs. Li et al. (2021) decomposed the crude oil price into multiple baseline models using variational mode decomposition and analyzed the complexity of each time series component, which was considered critical to selecting the most suitable base model to produce accurate forecasts.

As highlighted by Li et al. (2021), the fluctuations in the time series of crude oil prices can be influenced by underlying factors, including economic, political, and other irregular events. In fact, crude oil supply chains have been heavily affected by abnormal events, such as severe weather. The induced short-term market imbalances in turn affect the characteristics of crude oil prices, making them complex, nonlinear, and nonstationary. As a result, it is difficult to predict periodic fluctuations driven by unexpected events and changes in the price trend. Gao et al. (2017) studied the fluctuation patterns of the crude oil price, finding that each short-term fluctuation pattern has diverse dynamic properties. This finding was reflected in the diversity of the ensemble of statistical autoregressive models, which had completely different statistical characteristics for different short-term time series periodicities. Therefore, it was recommended to model the instable and non-linear crude oil prices by using different base models that consider diverse fluctuation patterns in various short-term oil price segments (Gao et al., 2017). To overcome this problem, we propose a blending ensemble learning model so that the crude oil price forecasts are made up of multiple diverse regression models, each of them trained on a small amount of crude oil price time series data. In fact, the use of homogeneous ensemble models that lack diversity in the forecasting methods can lead to underperformance if the base model type is not well suited to capture the complexity of crude oil price dynamics. Heterogeneous ensembles incorporate various types of base models, offering a more comprehensive approach to capturing the volatility and intricacies of crude oil prices. The heterogeneous ensemble is also typically more robust to noise and fluctuations. Therefore, heterogeneous ensembles generally outperform individual models and homogeneous ensembles, especially when dealing with complex and volatile systems like crude oil markets (Yuan et al., 2023).

3 Data

The two most commonly used data sets used in this study are Brent and WTI. It is worth noting that WTI was chosen as the data source because it is a globally recognized and traded benchmark for oil prices, making it directly relevant for market analyses and comparisons. The duration of the Brent crude oil price data was from May 20, 1987 to February 28, 2022. The period of the WTI data was from April 1, 1987 to February 28, 2022. The Brent data were taken from the U.S. Energy Information Administration,[1] and the WTI data were collected from the MarketWatch database (available at marketwatch.com). Data frequency was daily, but not seasonally adjusted. The total number of price observations was 9,104. The experimental data were then also converted from daily to weekly and monthly for the purpose of medium-term predictions. The numbers of observations and other descriptive statistics of

---

[1] https://www.eia.gov/todayinenergy/detail.php?id=51538.

Table 2 Descriptive statistics of Brent and WTI crude oil price data

	Brent_day	Brent_week	Brent_month	WTI_day	WTI_week	WTI_month
No. of observations	9104	1791	413	9095	1828	421
Maximum	143.95	141.07	132.72	145.31	142.52	133.88
Minimum	9.10	9.44	9.82	9.50	9.65	11.35
Mean	46.75	46.82	46.85	44.30	44.32	44.36
Std. dev	32.02	32.09	32.08	28.80	28.73	28.69
Variance	0.056	0.059	0.068	0.025	0.043	0.055
Skewness*	0.86	0.85	0.85	0.86	0.87	0.86
Kurtosis*	− 0.39	− 0.41	− 0.43	− 0.29	− 0.30	− 0.33

*The values were calculated before and after data normalization. In both cases the Skewness and Kurtosis remain unchanged

daily, weekly and monthly data are presented in Table 2. Figure 1 shows the daily, weekly, and monthly data for crude oil prices as a function of time.

Both data sets represent time series of crude oil prices. The descriptive statistics for both crude oil prices are presented in Table 2. The kurtosis of both datasets was negative, indicating that the distribution has lighter tails than the normal distribution. This means that the outlier problem is not serious. The value of the skewness of the Brent and WTI oil data ranged from 0.5 to 1, suggesting that the data were moderately positively skewed. In this study,
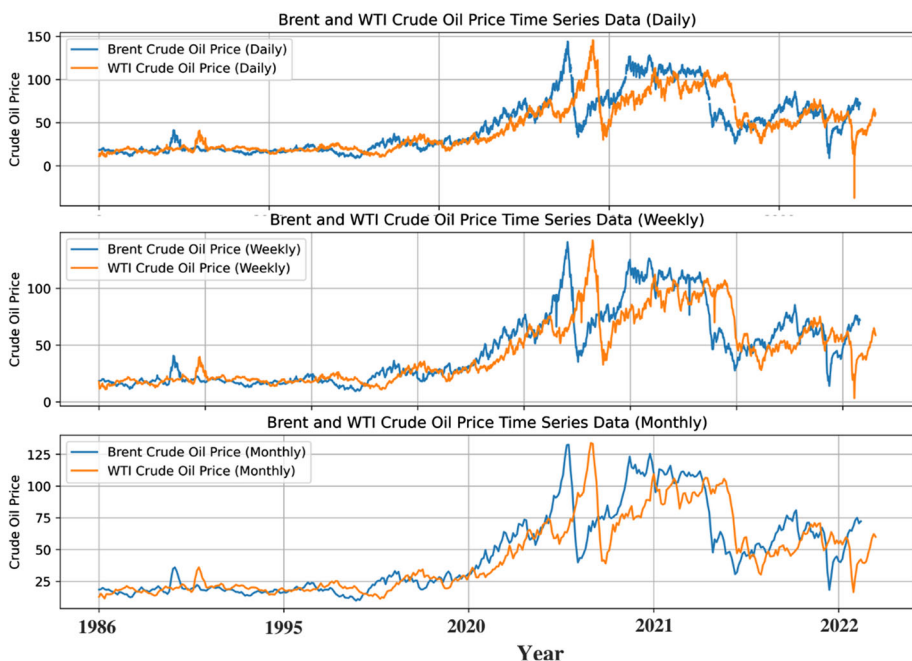
Fig. 1 Development of crude oil prices from 1986 to 2022

we used some machine learning algorithms that employ the Euclidean distance for model construction. To obtain a better forecasting performance of those base machine learning algorithms, the data were normalized in agreement with existing literature (Garbin et al., 2020). Using the normalization method, the data were dispersed less. Finally, the mean of the neighbors' values was used to handle the missing data.

4 Methods

The blending ensemble learning model referred to as LKDSR was used to predict daily, weekly, and monthly crude oil prices. The main advantages of this model are the computational efficiency of the blending approach and the heterogeneity of the base predictors, allowing us to model diverse oil price patterns.

4.1 Overview of the proposed methodology

The overview of the proposed methodology is depicted in Fig. 2. Brent and WTI crude oil daily data were collected and pre-processed to enhance the computational efficiency and model performance. Daily data were then converted to weekly and monthly data using the mode conversion module. Consistent with prior research (Herrera et al., 2018; Liang et al., 2023), the data sets pertaining to oil prices were partitioned using sequential validation methodology. Specifically, a 4:1 ratio was employed to split the data into training and testing
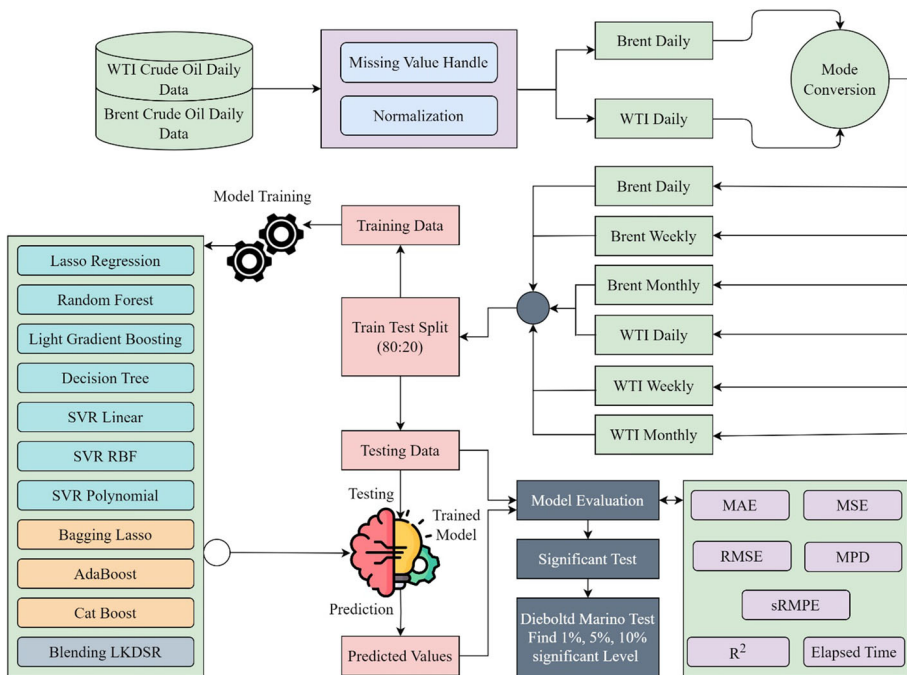

Fig. 2 Overview of the proposed methodology

subsets. The performance of the proposed and compared models was then evaluated using various error measures. To validate the statistical significance of our proposed model, we performed the DM test.

4.2 Data processing

After processing the missing values and normalizing the data using the MinMaxScaler function (to conduct feature normalization, scaling each individual feature to a pre-defined range between zero and one, thereby optimizing computational efficacy), both datasets underwent a mode conversion technique that converted the daily data to weekly and monthly data for short- to medium-term forecasts. Using the Python DateTime module, which allowed us to efficiently extract additional variables from the date variable, the daily data were converted to weekly and monthly data. Specifically, the mean of the daily data was taken to obtain the weekly (using the datetime.week function) and monthly (datetime.month function) data. Using Python DateTime module, we converted the daily data to weekly and monthly data, and then grouped them using the groupby function, as detailed in the pseudocode in Algorithm 1. We used the daily, weekly, and monthly data separately for training and testing. Each dataset was independent and had no effect on the training of the other datasets.

```
Input:  Daily Data
Output:  Weekly and Monthly Data
    1.  procedure data_convert ():
    2.     d = DATE
    3.     Week_Number = datetime.week(DATE)
    4.     Month_Number = datetime.month(DATE)
    5.     Weekly_data = groupby(Year, Week_Number).agg({Oil Price : mean})
    6.     Monthly_data = groupby(Year, Month_Number).agg({Oil Price : mean})
    7.  end procedure
```

Algorithm 1: Convert daily to weekly and monthly

4.3 Description of machine learning methods used

In the present study, individual and ensemble machine learning algorithms were used to forecast the price of crude oil. A brief description of the used algorithms is provided below.

4.3.1 Lasso regression

Lasso (Least Absolute Shrinkage and Selection Operator) is a modified linear regression performing both regularization and feature selection to enhance forecasting accuracy and intelligibility (Wang et al., 2011). The loss function for the lasso regression can be defined as follows:

$$\sum_{i=1}^{M}\left(y_i - \hat{y}_i\right)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{p} w_j \times x_{ij}\right)^2 + \lambda \sum_{j=0}^{p}|w_j| \qquad (1)$$

for some $t > 0$, $\sum_{j=0}^{p} |w_j| < t$, where w_j are model coefficients, t denotes a prespecified regularization parameter, and the shrinkage parameter λ controls the L_1 penalty. Lasso regression overcomes the disadvantages of ridge regression by restricting high coefficient values. Lasso sets the coefficient values to zero if they are irrelevant. Lasso regression introduces the magnitude instead of using the square of the coefficients. Lasso regression helps reduce overfitting and also performs feature selection, which makes the interpretation of this model easier.

While the application of Lasso regression techniques to the prediction of oil prices has historically been limited, emerging research demonstrates their robust performance relative to more complex statistical models. Hasan et al., (2023a, b) found that Lasso-based methods, known for their shrinkage properties, yielded markedly more accurate forecasts of oil price volatility compared to the Heterogeneous AutoRegressive model for Realized Variance (HAR-RV). Importantly, the empirical support for the superiority of the Lasso model remained consistent across an array of robustness checks, such as varying the sample sizes for estimating the shrinkage parameters, employing alternative estimation windows, and assessing performance over diverse out-of-sample periods. Boubaker et al. (2022) further substantiate the efficacy of Lasso regression in forecasting oil prices, attributing its performance to its capacity of handling the bias-variance tradeoff. This is facilitated through the shrinkage of coefficients, which effectively mitigates overfitting. Additionally, the Lasso model simplifies interpretability by setting certain coefficients to zero, thereby elucidating the most salient variables influencing oil price fluctuations.

4.3.2 Random forest regression

Random forest is one of the ensembles learning algorithms that is used in solving classification and regression problems (Hasan et al., 2023a, b). It combines tree predictors, with each decision tree including a random data vector while randomly determining a subsample of attributes. Hence, each node is partitioned by deploying random feature selection (Rabbi et al., 2022). Each predictor also randomly selects a training sample (Fig. 3). When comparing the error rate with the AdaBoost algorithm, the random forest is more robust in the presence of perturbations. The random forest algorithm is inherently flexible, simple, and robust. It provides excellent results even without adjusting the hyperparameters.

Mathematically, the prediction of the random forest regression model for a new input is expressed as:

$$\hat{Y} = \frac{1}{N} \sum_{i=1}^{N} Y_i \tag{2}$$

where $\hat{Y}$ is the predicted output, N is the number of trees in the forest, and Y_i represents the prediction from each individual tree. This ensemble approach results in a highly flexible and powerful regression model that can capture complex relationships between features and target variables while reducing the risk of overfitting often associated with single decision trees.

4.3.3 Light gradient boosting machine regression

Light gradient boosting machine (LGBM) is a tree-based gradient boosting method that works in a distributed manner and provides some important features; one of them is to
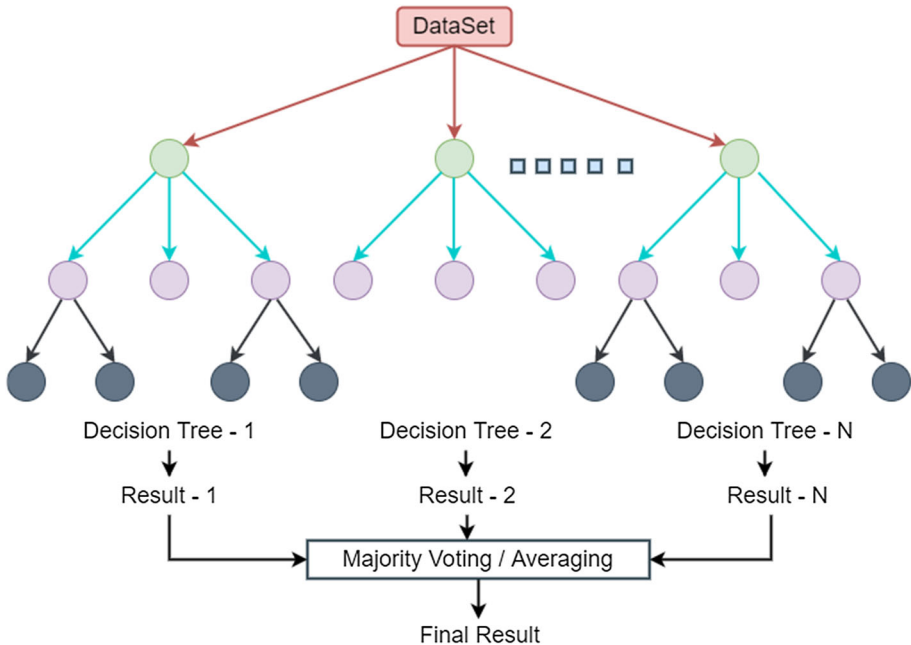
Fig. 3 The tree-based representation of random forest

increase the efficiency of the system and reduce memory usage. Efficient implementations of this ensemble decision tree are XGBoost and Pgbrt. The efficiency and scalability are unsatisfactory because the LGBM algorithm needs to search all the data in the instance, which is very time consuming. The two methods were introduced to solve this problem; one is exclusive feature bundling and the other is one side-sampling based on the gradient algorithm, which reduce the feature count and estimate the information gain using small data sample, respectively. Here is an overview of how LightGBM regression works:

Gradient boosting framework: LightGBM employs the gradient boosting framework, an ensemble learning method. This involves iteratively combining the predictions of several weak learners (typically decision trees) to form a robust predictive model.

Objective function: for regression tasks, LightGBM uses an objective function that calculates the difference between the actual values of the target variables and the error in the predicted values. Commonly used objective functions include Mean Squared Error (MSE) and Mean Absolute Error (MAE).

Tree building: unlike the traditional depth-first tree growth, LightGBM grows decision trees leaf-wise. This method prioritizes the development of leaves that lead to the largest decrease in the objective function, making the process faster and more memory-efficient.

Leaf-wise splitting: during tree growth, LightGBM selects the best feature and split point for each leaf node based on the gradient of the objective function. This allows the model to concentrate on areas of the data that are more challenging to predict, thus improving its accuracy.

Regularization: to avoid overfitting, LightGBM incorporates regularization techniques such as L1 (Lasso) and L2 (Ridge) regularization. The intensity of regularization can be adjusted using hyperparameters.

Gradient boosting and learning rate: LightGBM incrementally refines the model's predictions through gradient boosting. It starts with an initial prediction (often the mean of the target variables) and updates this with every new tree added. A learning rate hyperparameter controls the step size in each iteration.

Hyperparameter tuning: achieving optimal performance with LightGBM necessitates tuning hyperparameters like the number of trees (boosting rounds), tree depth, learning rate, and more. Methods like grid search or random search can help determine the best hyperparameters.

Prediction: once trained, the model can predict on new data by aggregating predictions from each individual tree using the ensemble method.

LightGBM is known for its speed and efficiency, making it an excellent choice for regression tasks, especially when dealing with large datasets or high-dimensional feature spaces (Sajid et al., 2023).

4.3.4 AdaBoost

AdaBoost (Adaptive Boosting) combines multiple weaker predictors into one strong regression model (Janssens et al., 2022; Shajalal et al., 2023). This algorithm operates according to the following steps. First, a dataset is provided at the beginning, and each data point is assigned an identical weight. Next, the data are given as input to the model and the erroneously predicted data instances are identified. The weight increments of these data instances are accumulated to ensure some changes in the model. Mathematically, the final prediction (Y) for a binary classification problem can be expressed as:

$$Y(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \tag{3}$$

where $Y(x)$ is the final prediction for input x, α_t represents the weight assigned to weak classifier $h_t(x)$, and T is the total number of weak classifiers. AdaBoost adjusts the weights and selects weak classifiers based on their ability to correct the mistakes of the previous classifiers, resulting in a strong ensemble model that excels at capturing complex decision boundaries.

4.3.5 Support vector regression

SVR allows us to flexibly define how much error we can accept in the model and helps us create a suitable function fit to the data (Abedin et al., 2018). The main function of SVR is to minimize the coefficients, more precisely, the coefficient vector and the L2-norm, rather than solely the squared error. The flexible nature of SVR is attributed to the kernel function, which implicitly transforms the data into a higher-dimensional feature space. The optimization problem in SVR can be expressed as:

$$\min_{w, b, \varepsilon, \varepsilon^*} \frac{1}{2}w^2 + C \sum_{i=1}^{n}(\varepsilon_i + \varepsilon_{i*}) \tag{4}$$

Subject to:

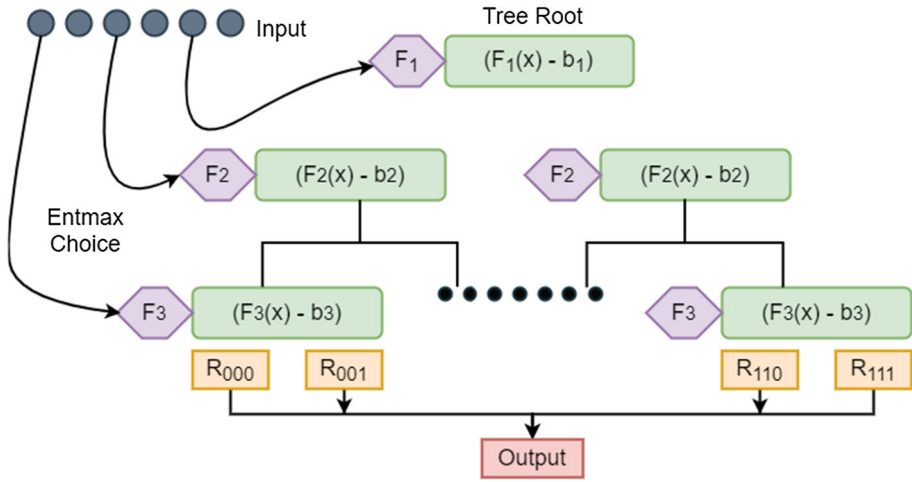$$y_i - (w.x_i + b) \leq \epsilon + \varepsilon_i \tag{5}$$

Fig. 4 Data modeling with CatBoost regressor

$$(w \cdot x_i + b) - y_i \leq \epsilon + \varepsilon_i \tag{6}$$

$$\varepsilon_i \varepsilon_{i*} \geq 0 \tag{7}$$

where w represents the weight vector, b is the bias term, x_i and y_i are the input and target values, ϵ is the tolerance margin, and C is the regularization parameter that balances margin size and error penalties. SVR is effective in capturing complex relationships in data, especially in cases where traditional regression techniques may not perform well.

4.3.6 CatBoost

CatBoost is one of the newest boosting ensemble machine learning models. This new open-source algorithm was developed in 2017 by a company named Yandex. As opposed to other boosting algorithms, CatBoost employs ordered boosting, an efficient enhancement of gradient boosting to address the issue of target leakage (Prokhorenkova et al., 2018). This is also effective when handling small datasets and categorical features, which can be replaced with one or more continuous values. As depicted in Fig. 4, the entmax activation function is employed as a soft variant of the splitting function, which is shared over the nodes at the same depth. The output is calculated as the weighted sum of the leaf node outputs.

4.3.7 Blending ensemble learning

In this study, we developed a blending ensemble learning model comprising five machine learning algorithms: linear regression, k-nearest neighbor (KNN) regression, decision tree regression, SVR, and ridge regression. This model, termed the LKDSR regression model, employs a second-order ensemble method known as blending.

While blending shares similarities with the stacking ensemble process, it possesses distinctive advantages. For instance, while stacking leverages out-of-fold predictions to train subsequent layers in the metamodel, blending uses a small validation set (comprising 10–15%
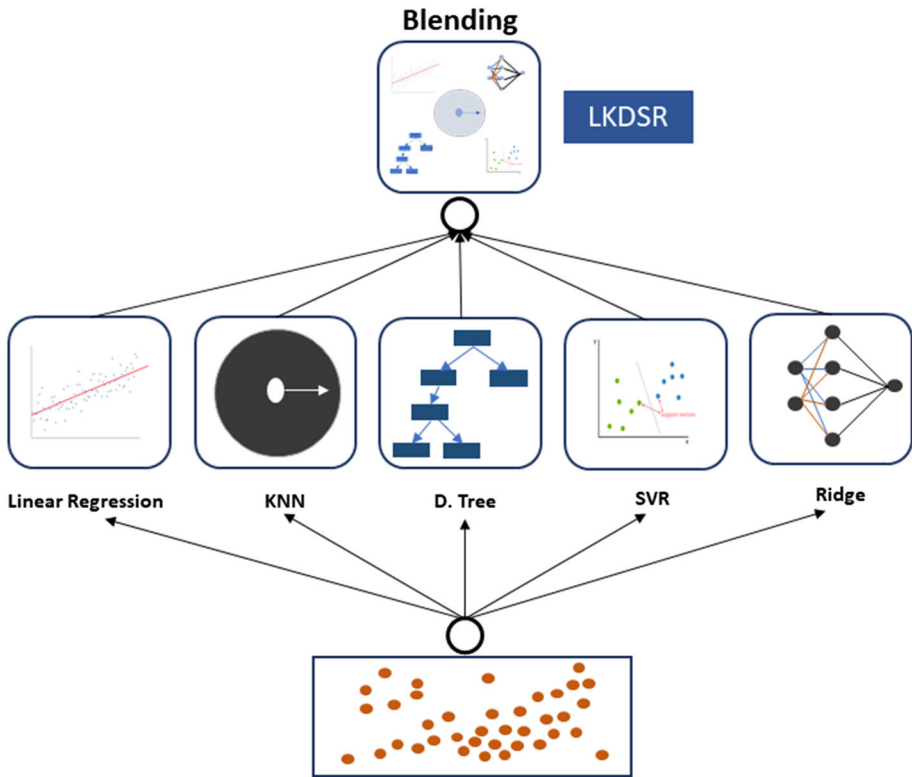
Fig. 5 Block diagram of the proposed LKDSR blending ensemble learning model

of the training data) for the same purpose. LKDSR integrates the mapping functions acquired from its member algorithms, as detailed in the workflow presented in Fig. 5.

The motivation to employ an ensemble over a singular model is rooted in the belief that ensemble models generally predict with greater accuracy and offer superior performance compared to individual ML models. Additionally, ensembles help decrease the spread of predictions, enhancing model reliability. The mapping functions from member algorithms merge to provide enhanced predictive capabilities. Our proposed ensemble model incorporates various methods:

- A general regression model: linear regression.
- Two distance-based ML models: KNN and SVR.
- A tree-based ML model: decision tree regression.
- A regularization technique: ridge regression.

The primary objective is to minimize data variance to optimize prediction accuracy. With time-series data, variance often escalates over time. Our model is designed to counteract this variance growth, thereby preventing overfitting during the final stages of training.

Firstly, the linear regression starts by predicting y from the independent variable x:

$$y = mx + c \tag{8}$$

Then the optimal k value for KNN is chosen from the data to find the best fitting line to predict the crude oil price. Here, we use Euclidean distance to measure the distance between the new data point y_i and the existing data point x_i:

$$\text{distance} = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \tag{9}$$

Here, distance refers to the distances between the p-dimensional feature vectors. The nearest neighbor of a vector $\vec{x}$ is the $\vec{x_i}$ closest to it. The k vectors of the k nearest neighbors are the $\vec{x}$ that are closest to $\vec{x}$. Keeping track of the indices of the neighbors, so we can write $NN(\vec{x}, j)$ for the index of the jth nearest neighbor of $\vec{x}$. The KNN estimate of the regression function is then the average value of the answer over the k nearest neighbors:

$$\hat{\mu}(\vec{x}) = \frac{1}{k}\sum_{j=1}^{k} y_{NN(\vec{x}, j)} \tag{10}$$

Then, we add the tree-based algorithm Decision Tree to explore all possible solutions. When KNN reduces the variance, the performance of DT is increased. Using a top-down, greedy search through the space of feasible branches without backtracking, the ID3 technique creates decision trees. The ID3 algorithm can produce a regression decision tree by replacing Information Gain with Standard Deviation Reduction. Mathematically, we can write residual as follows:

$$\varepsilon_i = y_i - \hat{y}_i \tag{11}$$

The residual sum of square is calculated by the following equation:

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{12}$$

Finally, the total RSS of each node is calculated by:

$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + \cdots + \varepsilon_n^2 \tag{13}$$

We subsequently incorporated SVR into the ensemble to derive predictions from the low-variance data produced by the DT. Time series data frequently display non-linearity that eludes linear models. In these instances, SVR, owing to SVM's capacity to address data non-linearity in regression tasks, proves effective for time series forecasting. SVR's primary goal is to approximate the regression function as outlined in form (8). Our goal is to find the best fitting line (which in the case of SVM is a hyperplane) that has the maximum number of data points. DT distributes the data points in such a way that we can easily get the best fit line for SVR and the error rate will be minimum then other algorithms.

Finally, we integrated the Ridge Regression regularization method into the ensemble pipeline. In this ensemble, we use L2 regularization that adds a penalty equal to the square of the magnitude of the coefficients. Ridge also adds bias to the estimators to reduce the standard error. Ridge reduces the variance by selecting the appropriate lambda.

The equation of Ridge is as follows:

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = RSS + \lambda\sum_{j=1}^{p}\beta_j^2 \tag{14}$$

Here, the value of RSS is calculated by DT above. The lambda is a tuning parameter, increasing the lambda will reduce the variance with a small bias. It also decreases the MSE and gives the best prediction.

Through this blending pipeline, Blending LKDSR reduces the variance and increases the accuracy of the prediction. Figure 5 shows an overview of the proposed model.

For a pseudo code representation of the ensemble algorithm, the following is an explanation of the algorithm.

Pseudocode of Blending Ensemble Algorithm

Algorithm: Blending LKDSR

Input

D: Time series crude oil price, $D = \{(x_i, y_i) | x_i \in X, y_i \in Y\}$

M: The number of regressors $(M_1, M_2, M_3, M_4, M_5)$

R: The parameter determining the proportion of samples that need to be replaced

L: The parameter used to control the distance from the synthetic samples to the training samples

Training Begin:

Operation: Prediction using ML Models *(D)*
for $i \longleftarrow$ *1 to M:*
    get a copy of the original training set, $D_i = D$
    get a number of training sample from the dataset and define it as P.
    for $j \longleftarrow$ *1 to P:*
        Train M_i model by P_j
    return new *data from the output of M_i*
End Loop
Regression Phase:
For a given x, the ensemble regressor $(M_1, M_2, M_3, M_4, M_5)$ predicts the final value y

4.4 Performance measures

In this study, MAE, MSE, RMSE, MPD, and sMAPE are used to evaluate forecasting performance as well as R^2 score.

The mean absolute error (MAE) is a statistic that evaluates the average error magnitude for a set of forecasted values. It is the average of the absolute differences between the forecast $\hat{y}_i$ and the actual observation y_i in the test sample, with all individual errors assigned with the same weigh:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|. \tag{15}$$

The mean square error (MSE) calculates the distances between the data instances and the regression function by squaring them to give more weight to larger differences:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{16}$$

The root mean square error (RMSE) is calculated as the standard deviation of the MSE residuals:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}. \tag{17}$$

The goodness of fit of a model is measured by its deviation. Comparing the result of our model to a baseline model is one approach to understand the amount of deviation. The loss due to the mean Poisson deviation (MPD) is such a regression performance measure, where the power parameter of $p = 1$ is equivalent to the Poisson deviation. If $\hat{y}_i$ is the predicted value for the i-th sample and y_i denotes the corresponding target value, then MPD is calculated as follows:

$$\text{MPD}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}} \begin{cases} (y_i - \hat{y}_i), & \text{for } p = 0 \text{(Normal)} \\ 2\left(y_i \log\left(\frac{\hat{y}_i}{y_i}\right) + \hat{y}_i - y_i\right), & \text{for } p = 1 \text{(Poisson)} \end{cases}. \tag{18}$$

Symmetric mean absolute percentage error (sMAPE) represents a percentage-based accuracy criterium. The absolute error is divided by the magnitude of the actual value to get the relative error. sMAPE has both a lower and an upper bound, unlike MAPE. It is scale-independent; therefore, it can be used to compare forecasting performance across datasets. It can be defined as follows:

$$\text{sMAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)}{(y_i + \hat{y}_i)/2}. \tag{19}$$

For linear regression problems, R^2 is used as metric of goodness of fit. This statistic represents the variance percentage for the dependent attribute that the independent attributes account for, measuring the strength of the association between the model outcome and the target values:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y}_i)^2}. \tag{20}$$

Finally, the elapsed time was measured as the total time taken by a model from the beginning to the end of the model calculation, that is, the elapsed time = end time – start time.

4.5 Significance test

A statistical test is a formal procedure to determine the best predictive model from a variety of models. In this study, we used one of the most commonly used methods, called the Diebold-Mariano (DM) test (Lago et al., 2021) to determine the significance level of our proposed model. The DM test is one the most commonly used techniques for establishing the statistical significance of forecasting accuracy disparities. It is an asymptotic Z-test (Diebold, 2015) of the hypothesis that the differential series loss mean is:

$$\Delta_k^{A, B} = L\left(\varepsilon_k^A\right) - L\left(\varepsilon_k^B\right), \tag{21}$$

where $L(\cdot)$ is the loss function and $\varepsilon_k^z = p_k - \hat{p}_k$ represents the forecasting error at time step k for model Z. For forecasting, we use $L(\varepsilon_k^Z) = |\varepsilon_k^Z|^P$ with $p = 1$ or 2, that is, the absolute

or squared errors, respectively. In this study, we consider the null hypothesis to be H_0: the loss difference of model A is less than or equal to that of model B. Here, if the hypothesis is rejected then model B significantly outperforms model A. In each hypothesis, test model B is our proposed LKDSR blending model.

5 Experimental results

5.1 Experimental setting

The hyperparameters of the forecasting models used, as obtained using the grid search procedure for the discrete values of each parameter, are presented in Table 3. The proposed blending LKDSR model is compared with methods used in earlier studies on crude oil price prediction, namely Lasso (Costa et al., 2021), SVR (Xie et al., 2006), AdaBoost (Guliyev & Mustafayev, 2022), regression tree (Chen & He, 2019), LGB (Gu et al., 2021), CatBoost (Jabeur et al., 2022), and random forest (Costa et al., 2021). We utilized the Scikit-learn library within the Python programming environment. All experimental evaluations detailed in this paper were conducted on a dedicated workstation with a 1.30 GHz CPU, 16 GB of RAM, and an Intel iRISxe graphics card. Additionally, all the experiments can be executed on Google Colab using any PC or mobile device with a stable internet connection.

The "Grid Search" method is employed for hyperparameter tuning to systematically identify the optimal set of hyperparameters for a machine learning model. It operates by creating a grid of hyperparameters, specifying their possible values or ranges, and then performing cross-validation on the model for each combination before training and evaluation. The most

Table 3 Values of hyperparameters for the forecasting models used

Model	Values of hyperparameters	Range of value search
Lasso	alpha = 0.01	alpha (0–1)
RF	n_estimators = 500, random_state = 42	n_estimators (100–1000) and random_state (10–100)
LGB	learning rate = 0.1, num_leaves = 24	learning_rate (0–1), num_leaves (1–100)
DT	random_state = 42	random_state (10–100)
B-Lasso	alpha = 0.01	alpha (0–1)
AdaBoost	n_estimators = 500, random_state = 42	n_estimators (100–1000) and random_state (10–100)
SVR_{Linear}	kernel = linear, gamma = auto, C = 100	gamma = Default, C (1–10)
SVR_{Rbf}	kernel = rbf, gamma = auto, C = 100	gamma = Default, C (1–10)
$SVR_{Polynomial}$	kernel = polynomial, gamma = auto, C = 100, degree = 3, coef = 1	gamma = Default, C (1–10), degree (2–10)
CatBoost	estimator = model_CBR, n_jobs = − 1, cv = 2, param_grid = parameters, learning rate = 0.05	learning rate (0.1–1)
LKDSR	kernel = linear, random_state = 42, n_estimators = 500, gamma = auto, C = 100	n_estimators (100–1000) and random_state (10–100), gamma = Default, C (1–10)

effective configuration is ascertained by the set of hyperparameters that yield the highest performance based on a specific evaluation metric. While Grid Search is a straightforward and comprehensive approach to finding the best hyperparameters, it can be computationally intensive, particularly when dealing with extensive parameter spaces.

5.2 Results of the Brent Crude oil price forecasting

The first result shows the performance of daily Brent crude oil forecasting using different machine learning methods. Table 4 shows that the forecasting performance of the LKDSR model is superior compared to the other models used in this analysis. The absolute and relative errors are relatively lower than those of the other models and a R^2 of 99% was achieved, suggesting that the proposed blending LKDSR model more accurately models short-term fluctuations in the data and provides more accurate daily forecasts of the Brent crude oil price than the other models. Although the elapsed time is not less than that of the other models, the elapsed time is still acceptable and competitive compared to the other models.

The actual vs. predicted curve of the models is shown in Fig. 6, with the blending LKDSR model curve being closer to the actual curve compared to the others. In contrast, SVR-based forecasting models were not effective, either significantly overestimating (SVR$_{Rbf}$) or underestimating (SVR$_{Polynomial}$) the predicted price.

Similarly, to daily forecasting, Tables 5 and 6 show the results for weekly and monthly prices of Brent crude oil, respectively. For the weekly Brent crude oil dataset, the proposed blending LKDSR model excelled in terms of all forecasting error measures, with errors approximately twice as large as for the daily prediction. Again, the SVR-based models performed poorly and only the boosting-based ensemble models (LBG and CatBoost) were competitive forecasting models.

Completely different behaviors of the models can be observed for the monthly prediction of Brent crude oil. Even though the proposed blending LKDSR performed best in terms of MAE, a significant decrease in forecasting accuracy is evident. Surprisingly, the SVR model with a polynomial kernel function had a performance similar to that for weekly predictions,

Table 4 Results for daily Brent crude oil dataset

Model	MAE	MSE	RMSE	MPD	sMAPE	R^2	Elapsed time
Lasso	0.0173	0.00051	0.0214	0.0014	5.2484	0.95	0.0039
Random forest	0.0108	0.00020	0.0142	0.0008	3.7294	0.98	**0.0011**
LGB	0.0073	**0.00009**	0.0099	**0.0005**	2.6474	**0.99**	0.0242
Regression tree	0.0107	0.00019	0.0139	0.0008	3.6696	0.98	0.0041
Bagging-Lasso	0.0174	0.00046	0.0215	0.0014	5.2709	0.95	0.0051
AdaBoost	0.0094	0.00016	0.0125	0.0006	3.3489	0.98	0.0057
SVR$_{Linear}$	0.0174	0.00042	0.0206	0.0017	6.2410	0.95	0.0103
SVR$_{Rbf}$	0.0166	0.00039	0.0198	0.0017	6.0295	0.96	0.0031
SVR$_{Polynomial}$	0.0265	0.00098	0.0313	0.0032	8.2504	0.90	0.0103
CatBoost	0.0074	0.00010	0.0101	**0.0005**	2.6944	**0.99**	0.0039
LKDSR	**0.0070**	**0.00009**	**0.0097**	**0.0005**	**2.5878**	**0.99**	0.0048

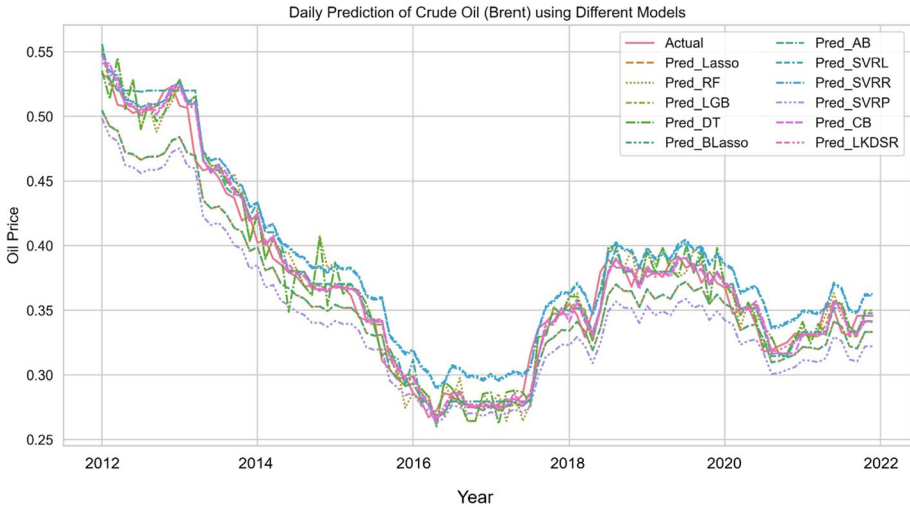The best performance is marked in bold

Fig. 6 Actual vs. predicted curve of daily prediction of Brent crude oil price

Table 5 Results for the weekly Brent crude oil data set

Model	MAE	MSE	RMSE	MPD	sMAPE	R^2	Elapsed time
Lasso	0.0205	0.00067	0.0259	0.0022	6.3856	0.93	0.0081
Random forest	0.0259	0.00164	0.0405	0.0042	8.0664	0.83	**0.0007**
LGB	0.0161	0.00046	0.0216	0.0017	5.4916	0.93	0.0204
Regression tree	0.0214	0.00075	0.0274	0.0026	7.0691	0.92	0.0022
Bagging-Lasso	0.0209	0.00069	0.0263	0.0022	6.4644	0.93	0.0041
AdaBoost	0.0258	0.00112	0.0335	0.0032	7.7629	0.89	0.0022
SVR$_{Linear}$	0.0286	0.00124	0.0352	0.0046	9.7553	0.87	0.0046
SVR$_{Rbf}$	0.0245	0.00098	0.0314	0.0039	8.6371	0.90	0.0061
SVR$_{Polynomial}$	0.0373	0.00185	0.0430	0.0060	11.8254	0.81	0.0018
CatBoost	0.0157	0.00045	0.0211	0.0016	5.4031	0.95	0.0053
LKDSR	**0.0143**	**0.00038**	**0.0194**	**0.0014**	**5.0393**	**0.96**	0.0177

The best performance is marked in bold

indicating strong polynomial relationships in the univariate monthly time series of the Brent crude oil price. Hence, the proposed blending LKDSR model is much less effective.

Likewise, Figs. 7 and 8 depict the weekly and monthly forecasts using the predictions obtained from the compared models. For the weekly forecasts (Fig. 7), we can observe a similar behavior to the daily prices, and most models overestimate the actual price of Brent crude oil. The results in Fig. 8 show that it was more difficult to predict the medium-term monthly prices, and most models did not react promptly to the change in the trend of the crude oil price.

To verify the statistical significance of the proposed model, the DM test was conducted, and the significance level was calculated for RMSE over the individual errors obtained for

Table 6 Results for the monthly Brent crude oil data set

Model	MAE	MSE	RMSE	MPD	sMAPE	R^2	Elapsed time
Lasso	0.0382	0.00237	0.0487	0.0078	11.8553	0.78	0.0111
Random forest	0.0500	0.00467	0.0683	0.0141	15.3919	0.57	0.0008
LGB	0.0385	0.00283	0.0532	0.0096	12.4335	0.74	0.0348
Regression tree	0.0478	0.00402	0.0634	0.0128	15.0064	0.63	0.0048
Bagging-Lasso	0.0383	0.00238	0.0487	0.0078	11.8612	0.78	0.0082
AdaBoost	0.0413	0.00339	0.0583	0.0111	13.1728	0.69	0.0086
SVR_{Linear}	0.0379	0.00236	0.0486	0.0083	12.2966	0.78	0.0039
SVR_{Rbf}	0.0391	0.00247	0.0497	0.0093	13.0353	0.77	0.0022
$SVR_{Polynomial}$	0.0363	**0.00225**	**0.0475**	**0.0073**	**11.1847**	**0.79**	**0.0001**
CatBoost	0.0389	0.00295	0.0543	0.0097	12.5038	0.73	0.0007
LKDSR	**0.0360**	0.00233	0.0483	0.0082	11.7096	**0.79**	0.0053

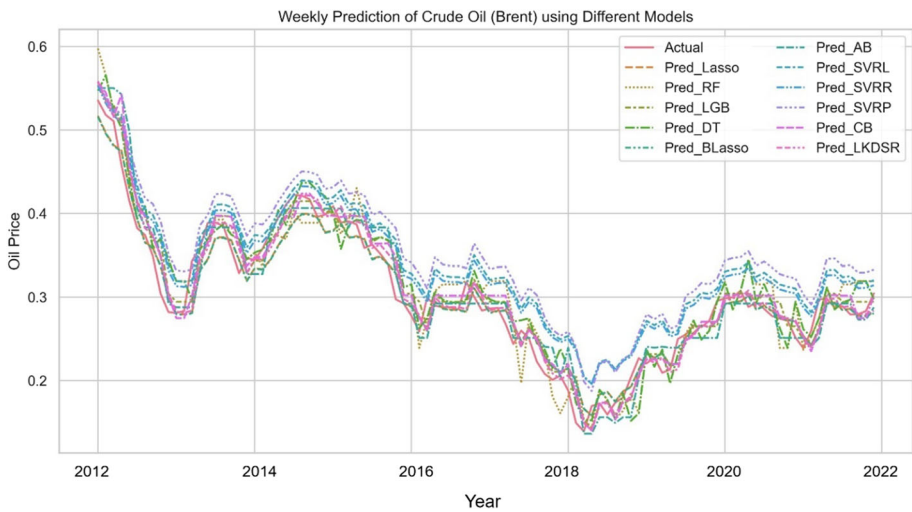The best performance is marked in bold

Fig. 7 Actual vs. predicted curve of the weekly prediction of the Brent crude oil price

the test data instances. To find significant contributions of our proposed methodology, we applied the DM test to the daily, weekly, and monthly prices of Brent crude oil. The DM values for the models compared with our blending LKDSR model are shown in Table 7. For the daily and weekly forecasts of the Brent crude oil price, our proposed model significantly outperformed the other models used in this analysis. In contrast, the SVR with polynomial kernel function was superior for monthly data, with the blending LKDSR model ranked second and significantly outperforming the remaining forecasting models.
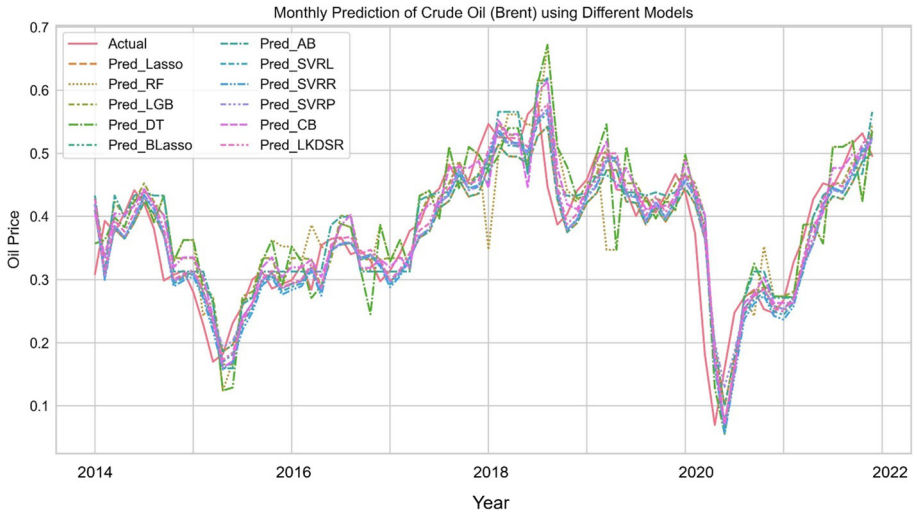
Fig. 8 Actual vs. predicted curve of the monthly prediction of the Brent crude oil price

Table 7 Results of the DM statistical test for Brent crude oil price—LKDSR blending model vs. compared models

Model	Brent daily (DM value)	Brent weekly (DM value)	Brent monthly (DM value)
Lasso	19.01217*	5.303314*	0.194063***
Random forest	13.40259*	3.608193*	2.908655*
LGB	4.273887*	2.979493*	2.066468**
Regression tree	13.67913*	7.043103*	2.477125**
Bagging-Lasso	19.06145*	5.443024*	0.208523***
AdaBoost	10.19771*	7.477397*	2.763016***
SVR$_{Linear}$	21.85069*	7.738753*	0.223951***
SVR$_{Rbf}$	19.79711*	5.858687*	0.975974***
SVR$_{Polynomial}$	23.50757*	11.35995*	− 0.65528***
CatBoost	5.696250*	2.710647*	2.017943**

The DM test statistics examine the accuracy of our blending LKDSR model against the benchmarks for the prediction of the price of Brent crude oil

***, **, *represent statistical significance at $P < 10\%$, $P < 5\%$, and $P < 1\%$, respectively

5.3 Results of WTI Crude oil price forecasting

Similar to the Brent oil price, Table 8 shows a considerable difference between the forecasting performance for daily, weekly, and monthly WTI forecasting horizons. This can be attributed to the fact that daily data often captures more granular market movements, allowing models to fine-tune their predictive ability based on richer data sets. In addition, monthly forecasts inherently incorporate more complexity and uncertainty due to the longer forecasting horizon, which can be influenced by a variety of unpredictable factors such as geopolitical events and economic uncertainty. Compared to the Brent oil price, the blending LKDSR model

Table 8 Results for the WTI crude oil data set

Model	MAE	MSE	RMSE	MPD	sMAPE	R^2	Elapsed time
Daily prediction							
Lasso	0.0381	0.00275	0.0524	0.0049	6.9644	0.75	0.0024
Random forest	0.0083	0.00016	0.0128	0.0006	1.7442	0.98	0.0021
LGB	0.0056	0.00011	0.0103	0.0004	1.2206	**0.99**	0.0088
Regression tree	0.0084	0.00017	0.0132	0.0006	1.7464	0.98	0.0112
Bagging-Lasso	0.0383	0.00281	0.0527	0.0049	7.0129	0.75	0.0074
AdaBoost	0.0115	0.00026	0.0160	0.0007	2.3833	0.98	0.0192
SVR_{Linear}	0.0277	0.00097	0.0311	0.0024	5.7825	0.91	0.0021
SVR_{Rbf}	0.0275	0.00096	0.0309	0.0023	5.7519	0.91	0.0169
$SVR_{Polynomial}$	0.0158	0.00049	0.0221	0.0014	3.5150	0.96	0.0034
CatBoost	0.0057	0.00011	0.0103	0.0004	1.2345	**0.99**	**0.0003**
LKDSR	**0.0054**	**0.00010**	**0.0101**	**0.0002**	**1.2057**	**0.99**	0.0075
Weekly prediction							
Lasso	0.0256	0.00121	0.0348	0.0035	7.1489	0.91	0.0109
Random forest	0.0199	0.00072	0.0268	0.0027	6.4048	0.95	**0.0013**
LGB	0.0148	0.00041	0.0202	**0.0018**	5.0459	0.97	0.0202
Regression tree	0.0202	0.00071	0.0267	0.0027	6.5214	0.95	0.0039
Bagging-Lasso	0.0262	0.00126	0.0355	0.0036	7.2751	0.91	0.0031
AdaBoost	0.0216	0.00081	0.0284	0.0027	6.6569	0.94	0.0162
SVR_{Linear}	0.0295	0.00122	0.0349	0.0045	9.3198	0.91	0.0109
SVR_{Rbf}	0.0308	0.00129	0.0359	0.0047	9.6443	0.91	**0.0013**
$SVR_{Polynomial}$	0.0364	0.00169	0.0411	0.0059	11.264	0.88	0.0082
CatBoost	0.0146	0.00041	0.0202	0.0019	4.9814	0.97	0.0107
LKDSR	**0.0128**	**0.00033**	**0.0182**	0.0031	**4.8758**	**0.98**	0.0123
Monthly prediction							
Lasso	0.0383	0.00226	0.0476	**0.0080**	12.8976	**0.87**	0.0172
Random forest	0.0462	0.00370	0.0608	0.0131	15.4885	0.78	0.0137
LGB	0.0373	0.00257	0.0506	0.0097	13.0653	0.85	0.0314
Regression tree	0.0478	0.00405	0.0637	0.0152	16.5186	0.76	0.0053
Bagging-Lasso	0.0383	0.00226	0.0476	**0.0080**	12.8907	**0.87**	0.0176
AdaBoost	0.0363	0.00235	0.0484	0.0089	12.7037	0.86	0.0123
SVR_{Linear}	0.0378	0.00223	0.0472	0.0098	13.9456	**0.87**	0.0047
SVR_{Rbf}	0.0398	0.00246	0.0496	0.0125	15.1690	0.86	0.0038
$SVR_{Polynomial}$	0.0408	0.00249	0.0499	0.0094	14.4076	0.85	0.0005
CatBoost	0.0367	0.00267	0.0517	0.0105	13.0674	0.84	**0.0004**
LKDSR	**0.0351**	**0.00219**	**0.0468**	0.0088	**12.6189**	**0.87**	0.0085

The best performance is marked in bold

performed better for daily and weekly WTI data. In addition, unlike the Brent crude oil price, our model excelled for the monthly WTI data. From Table 8, it can also be noted that the boosting-based forecasting models also provide solid performance, while Lasso-based models give sound forecasting performance only for monthly time series data.

Figures 9, 10 and 11 also show similar patterns observed above for the Brent crude oil price. Again, all the models tested had difficulties responding to strong price fluctuations in a timely manner.
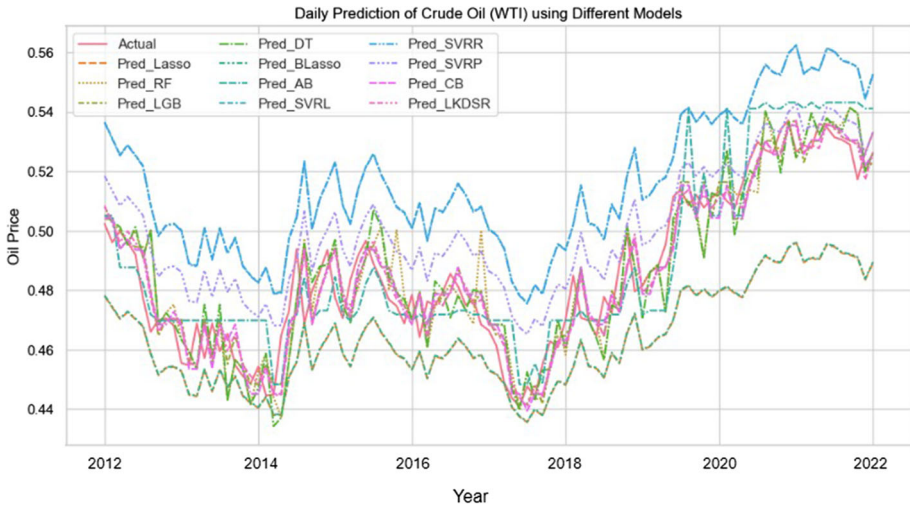
Fig. 9 Actual vs. predicted curve of daily prediction of the WTI price
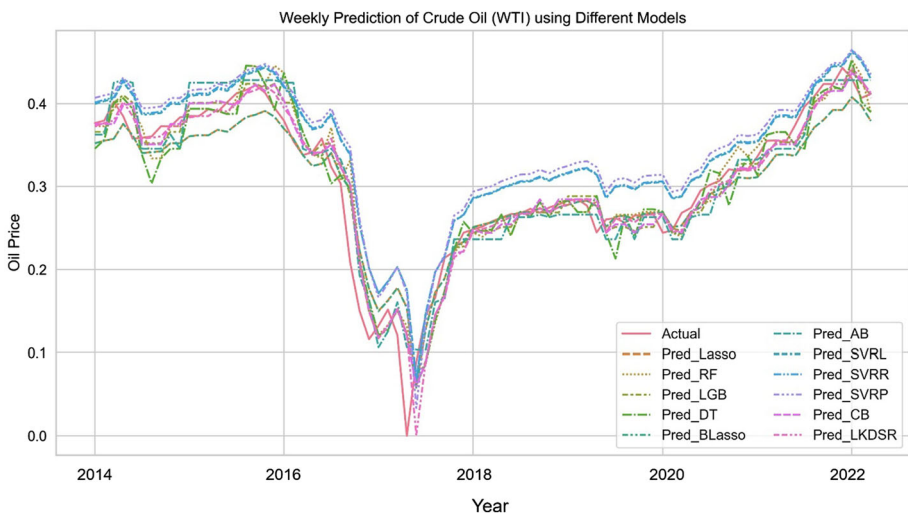
Fig. 10 Actual vs. predicted curve of the weekly prediction of the WTI price
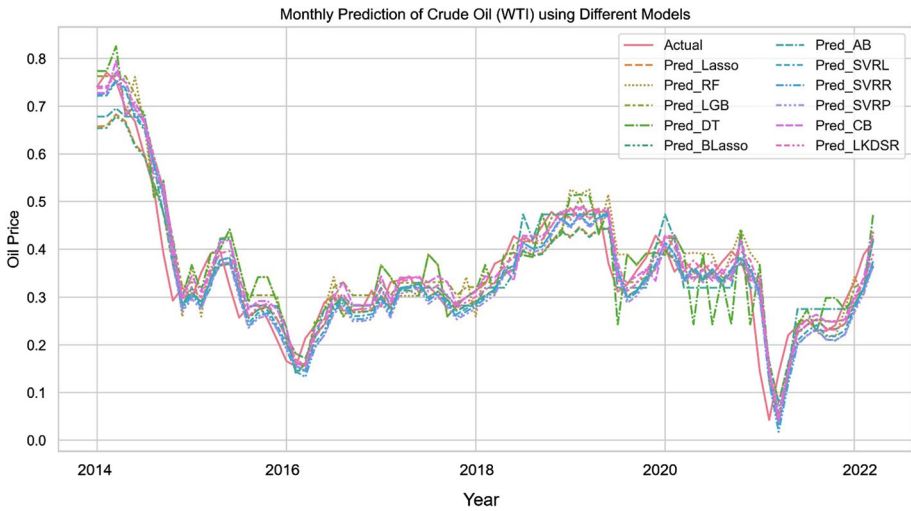
Fig. 11 Actual vs. predicted curve of the monthly prediction of the WTI price

The results of the DM statistical tests are presented in Table 9. As presented in Table 9, a significant difference was found between the forecasting performance of the blending LKDSR model and those of the models compared. The tests revealed significant differences for all the forecasting horizons examined. Strong evidence of the superiority of the proposed blending approach was found even for the monthly forecasts. Overall, these results suggest that the forecasting performance of the blending LKDSR model is robust to the granularity of WTI time series.

To demonstrate the superiority of our proposed blending ensemble algorithm over other ensemble strategies, we constructed three distinct ensemble models using stacking, majority

Table 9 Results of the DM statistical test for the WTI price—LKDSR blending model vs. compared models

Model	WTI daily (DM Value)	WTI weekly (DM Value)	WTI monthly (DM Value)
Lasso	14.13154*	4.917655*	0.169650***
Random forest	0.86794***	6.488912*	4.729863*
LGB	- 0.83161***	2.407989**	2.109551**
Regression tree	1.10766***	5.908309*	4.142767*
Bagging-Lasso	14.16250*	4.979688*	0.174012***
AdaBoost	3.50721*	5.482341*	0.633890***
SVR$_{Linear}$	38.13681*	9.492517*	0.199953***
SVR$_{Rbf}$	37.11603*	10.133650*	1.035432***
SVR$_{Polynomial}$	12.31610*	12.227580*	1.082033***
CatBoost	- 0.79950***	2.336348**	2.515218**

*At $P < 1\%$
**At $P < 5\%$
***Significant at $P < 10\%$

Table 10 Comparison of proposed blending algorithm with stacking, majority voting and averaging ensemble techniques

Model	MAE	MSE	RMSE	MPD	sMAPE	R^2
	Brent daily prediction					
Averaging	0.0129	0.00046	0.0146	**0.0003**	2.4621	0.90
Voting	0.0066	0.00039	0.0153	0.0004	2.7160	0.92
Stacking	**0.0057**	0.00013	0.0105	**0.0003**	**1.2066**	0.98
LKDSR	0.0070	**0.00009**	**0.0097**	0.0005	2.5878	**0.99**
	WTI daily prediction					
Averaging	0.0145	0.00880	0.0254	0.0030	5.3259	0.86
Voting	0.0064	0.00057	0.0225	0.0033	5.1624	0.88
Stacking	0.0132	0.00037	0.0200	0.0029	4.8878	0.97
LKDSR	**0.0054**	**0.00010**	**0.0101**	**0.0002**	**1.2057**	**0.99**

The best performance is marked in bold

voting, and averaging techniques. These models were applied to both the Brent and WTI crude oil datasets. While many of the ensemble algorithms showcased robust performance, our proposed blending ensemble consistently surpassed them. This superior performance is underscored by the R^2 values, suggesting that the algorithms are highly compatible with the dataset.

Although the stacking ensemble displayed performance on par with the blending ensemble, our blending algorithm, LKDSR, prevailed in most scenarios. The advantage of the stacking approach lies in its two-level system, where base models feed predictions to a meta-model. This setup fosters diverse model combinations but necessitates significant computational resources and intricate tuning. In contrast, blending employs a more straightforward strategy, eschewing a meta-model and directly amalgamating base model predictions through methods like averaging. While blending is more streamlined in implementation and requires fewer resources, it is usually tailored for a more homogenous set of base models. The decision between stacking and blending hinges on the specific problem, dataset, and computational resources at hand. Stacking offers versatility and complexity, whereas blending presents a more straightforward and resource-conservative alternative. The results can be found in Table 10.

6 Discussion

The main inspiration for our blending LKDSR forecasting model was to fully exploit state-of-the-art machine learning-based models by combining them as a computationally efficient machine learning ensemble that captures the short-term oil price imbalances. To determine the efficacy of the blending LKDSR model, we can compare our results with those obtained in the existing literature (Table 2). The comparison with various studies clearly indicates the superiority of our proposed blending ensemble model over existing solutions. The performance of the proposed blending LKDSR model in terms of error measures was outstanding, as compared with the relative error measures in Table 1.

This study appears to be the first to demonstrate the effects of using a blending ensemble learning approach in forecasting crude oil prices. Notably, the blending ensemble learning approach consistently outperformed individual models, aligning with previous findings on ensemble learning (Busari & Lim, 2021; Zhao et al., 2017). Importantly, our proposed LKDSR exhibited lower error rates in WTI daily forecasting, with MAE = 0.0054 and RMSE = 0.0101. This performance surpasses those of deep learning-based ensemble approaches, specifically the Bagging approach (Zhao et al., 2017) which had an RMSE of 4.9, and AdaBoost (Busari & Lim, 2021) which recorded an MAE of 1.42 and RMSE of 2.46.

The lower error rate and high accuracy of the proposed model is promising and potentially valuable for policy makers and shareholders in making future decisions. The oil market is highly volatile in the short term, but the proposed model allowed us to capture the nature of these short-term fluctuations. The diversity of the base forecasting models used in our ensemble approach considered various fluctuation patterns in short-term crude oil price segments. These findings are in line with those provided by Gao et al. (2017) and Li et al. (2021). The observed increase in prediction performance for the daily oil price could be interpreted as confirmation of the feasibility of modelling these fluctuations using the blending approach.

Interestingly, there are also differences in the forecasting performance for the two crude oil prices, Brent and WTI. The results in terms of the error criteria showed that more accurate forecasts were obtained for the WTI crude oil prices. Similar trends have been reported by Nademi and Nademi (2018) in their work on predicting the prices of crude oil by using semiparametric Markov models. Similarly, Karasu and Altan (2022) achieved better LSTM performance for the WTI crude oil prices by.

The price of crude oil is very important in energy policy. On a daily basis, various renewable energy sources are being developed that have a strong complex relationship with crude oil. When the price and supply of other energy sources change, this has an impact on the oil price market, making this market uncertain and affecting investor behavior. The present findings might help solve the problem of handling this uncertainty in short-term decision making. In further research, it would therefore be interesting to evaluate the effectivity of the produced crude oil forecasts by assessing the performance of the crude oil portfolio.

7 Conclusions

In this study, we have outlined the blending ensemble learning model combining five machine learning regression methods. The results showed that our blending LKDSR model has better forecasting performance with respect to various error criteria compared to the benchmark methods used in earlier research. The performance of our proposed model is also robust to the granularity of oil time series, which is important for industry production decisions. The results of this study can also guide stakeholders in finding the best investment plan that would yield profits during the oil crisis.

This research also highlights a number of issues that need further investigation. For oil markets, our proposed machine learning strategy is shown to be highly effective in modelling diverse fluctuation patterns. It would be interesting to see how effective it is in different commodity and financial markets, such as predicting stock prices, predicting foreign exchange rates, and predicting the prices of precious metals. For instance, there has long been a boost to forecasting oil in that one can also infer for gas, but these forecasting models broke down with the decoupling of oil and gas prices (e.g., Batten et al. (2017)). Some limitations should be considered. The present study has only investigated univariate crude oil time series prices.

We have not included other elements such as environmental variables, macroeconomic factors, and foreign market behavior because our primary focus was on univariate time series oil price forecasting patterns. Future research will develop forecasting models that will be informative for decision makers in estimating the impact of political and economic events that will evolve with the price of Brent and WTI oil on economic policies. In fact, more determinants could be included in the proposed forecast model in further studies, including the market sentiment indicators extracted from textual data such as news articles. This would allow for a more comprehensive representation of the multifaceted drivers influencing crude oil prices. In a similar way, the EIA's projections for the price of Brent crude oil can be used as a trend indicator for future prices. Thus, the forecasting power of the model could be improved, which opens the possibility of further research in the future. Similarly, a wider set of diverse machine learning methods could be considered in future research and ensemble selection (Lessmann et al., 2021) could be applied to select the best subset for the blending ensemble.

Declarations

Conflict of interest Disclosure of potential conflicts of interest: no conflict of interest exists.

Ethical approval Research involving human participants and/or animals: this article does not contain any studies with human participants performed by any of the authors.

Informed consent Not applicable.

References

Abedin, M. Z., Chi, G., Colombage, S., & Moula, F. E. (2018). Credit default prediction using a support vector machine and a probabilistic neural network. *Journal of Credit Risk, 14*(2), 1–27.

Abedin, M. Z., Moon, M. H., Hassan, M. K., & Hajek, P. (2021). Deep learning-based exchange rate prediction during the COVID-19 pandemic. *Annals of Operations Research*, 1–52.

Asai, M., Gupta, R., & McAleer, M. (2020). Forecasting volatility and co-volatility of crude oil and gold futures: Effects of leverage, jumps, spillovers, and geopolitical risks. *International Journal of Forecasting, 36*(3), 933–948.

Bakas, D., & Triantafyllou, A. (2020). Commodity price volatility and the economic uncertainty of pandemics. *Economics Letters, 193*, 109283.

Batten, J. A., Ciner, C., & Lucey, B. M. (2017). The dynamic linkages between crude oil and natural gas markets. *Energy Economics, 62*, 155–170.

Behmiri, N. B., & Manso, J. R. P. (2013). How crude oil consumption impacts on economic growth of Sub-Saharan Africa? *Energy, 54*, 74–83.

Bisoi, R., Dash, P. K., & Mishra, S. P. (2019). Modes decomposition method in fusion with robust random vector functional link network for crude oil price forecasting. *Applied Soft Computing, 80*, 475–493.

Boubaker, S., Liu, Z., & Zhang, Y. (2022). Forecasting oil commodity spot price in a data-rich environment. *Annals of Operations Research*. https://doi.org/10.1007/s10479-022-05004-8

Bristone, M., Prasad, R., & Abubakar, A. A. (2020). CPPCNDL: Crude oil price prediction using complex network and deep learning algorithms. *Petroleum, 6*(4), 353–361.

Busari, G. A., & Lim, D. H. (2021). Crude oil price prediction: A comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance. *Computers & Chemical Engineering, 155*, 107513.

Cen, Z., & Wang, J. (2019). Crude oil price prediction model with long short-term memory deep learning based on prior knowledge data transfer. *Energy, 169*, 160–171.

Cerqueti, R., & Fanelli, V. (2021). Long memory and crude oil's price predictability. *Annals of Operations Research, 299*(1), 895–906.

Chang, L., Baloch, Z. A., Saydaliev, H. B., Hyder, M., & Dilanchiev, A. (2022). Testing oil price volatility during Covid-19: Global economic impact. *Resources Policy, 78*, 102891.

Chen, E., & He, X. J. (2019). Crude oil price prediction with decision tree based regression approach. *Journal of International Technology and Information Management, 27*(4), 2–16.

Chen, Y., He, K., & Tso, G. K. (2017). Forecasting crude oil prices: A deep learning-based model. *Procedia Computer Science, 122*, 300–307.

Chen, Y. C., & Huang, W. C. (2021). Constructing a stock-price forecast CNN model with gold and crude oil indicators. *Applied Soft Computing, 112*, 107760.

Chen, Z. Y. (2022). A computational intelligence hybrid algorithm based on population evolutionary and neural network learning for the crude oil spot price prediction. *International Journal of Computational Intelligence Systems, 15*(1), 68.

Costa, A. B. R., Ferreira, P. C. G., Gaglianone, W. P., Guillén, O. T. C., Issler, J. V., & Lin, Y. (2021). Machine learning and oil price point and density forecasting. *Energy Economics, 102*, 105494.

Diebold, F. X. (2015). Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of Diebold–Mariano tests. *Journal of Business & Economic Statistics, 33*(1), 1–1.

Escribano, Á., & Wang, D. (2021). Mixed random forest, cointegration, and forecasting gasoline prices. *International Journal of Forecasting, 37*(4), 1442–1462.

Gao, X., Fang, W., An, F., & Wang, Y. (2017). Detecting method for crude oil price fluctuation mechanism under different periodic time series. *Applied Energy, 192*, 201–212.

Garbin, C., Zhu, X., & Marques, O. (2020). Dropout vs. batch normalization: An empirical study of their impact to deep learning. *Multimedia Tools and Applications, 79*(19), 12777–12815.

Ghaffari, A., & Zare, S. (2009). A novel algorithm for prediction of crude oil price variation based on soft computing. *Energy Economics, 31*(4), 531–536.

Gu, Q., Chang, Y., Xiong, N., & Chen, L. (2021). Forecasting Nickel futures price based on the empirical wavelet transform and gradient boosting decision trees. *Applied Soft Computing, 109*, 107472.

Guliyev, H., & Mustafayev, E. (2022). Predicting the changes in the WTI crude oil price dynamics using machine learning models. *Resources Policy, 77*, 102664.

Hao, M., Cao, W. H., Liu, Z. T., Wu, M., & Xiao, P. (2020). Visual-audio emotion recognition based on multi-task and ensemble learning with multiple features. *Neurocomputing, 391*, 42–51.

Hasan, M., Das, U., Datta, R. K., & Abedin, M. Z. (2023a). Model development for predicting the crude oil price: Comparative evaluation of ensemble and machine learning methods. *Novel financial applications of machine learning and deep learning: Algorithms, product modeling, and applications* (pp. 167–179). Springer.

Hasan, M., Marjan, M. A., Uddin, M. P., Afjal, M. I., Kardy, S., Ma, S., & Nam, Y. (2023b). Ensemble machine learning-based recommendation system for effective prediction of suitable agricultural crop cultivation. *Frontiers in Plant Science,* 14.

Herrera, A. M., Hu, L., & Pastor, D. (2018). Forecasting crude oil price volatility. *International Journal of Forecasting, 34*(4), 622–635.

Huang, L., & Wang, J. (2018). Global crude oil price prediction and synchronization-based accuracy evaluation using random wavelet neural network. *Energy, 151*, 875–888.

Ibrahim, B. A., Elamer, A. A., & Abdou, H. A. (2022). The role of cryptocurrencies in predicting oil prices pre and during COVID-19 pandemic using machine learning. *Annals of Operations Research*. https://doi.org/10.1007/s10479-022-05024-4

Jabeur, S. B., Mefteh-Wali, S., & Viviani, J. L. (2022). Forecasting gold price with the XGBoost algorithm and SHAP interaction values. *Annals of Operations Research*, 1–21.

Janssens, B., Bogaert, M., Bagué, A., & Van den Poel, D. (2022). B2Boost: Instance-dependent profit-driven modelling of B2B churn. *Annals of Operations Research*. https://doi.org/10.1007/s10479-022-04631-5

Karasu, S., & Altan, A. (2022). Crude oil time series prediction model based on LSTM network with chaotic Henry gas solubility optimization. *Energy, 242*, 122964.

Lago, J., Marcjasz, G., De Schutter, B., & Weron, R. (2021). Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. *Applied Energy, 293*, 116983.

Lessmann, S., Haupt, J., Coussement, K., & De Bock, K. W. (2021). Targeting customers for profit: An ensemble learning framework to support marketing decision-making. *Information Sciences, 557*, 286–301.

Li, R., Hu, Y., Heng, J., & Chen, X. (2021). A novel multiscale forecasting model for crude oil price time series. *Technological Forecasting and Social Change, 173*, 121181.

Li, X., Shang, W., & Wang, S. (2019). Text-based crude oil price forecasting: A deep learning approach. *International Journal of Forecasting, 35*(4), 1548–1560.

Li, Y., & Pan, Y. (2022). A novel ensemble deep learning model for stock prediction based on stock prices and news. *International Journal of Data Science and Analytics, 13*(2), 139–149.

Liang, X., Luo, P., Li, X., Wang, X., & Shu, L. (2023). Crude oil price prediction using deep reinforcement learning. *Resources Policy, 81*, 103363.

Lu, Q., Sun, S., Duan, H., & Wang, S. (2021). Analysis and forecasting of crude oil price based on the variable selection-LSTM integrated model. *Energy Informatics, 4*(2), 1–20.

Moshiri, S., & Foroutan, F. (2006). Forecasting nonlinear crude oil futures prices. *The Energy Journal, 27*, 81–95.

Mostafa, M. M., & El-Masry, A. A. (2016). Oil price forecasting using gene expression programming and artificial neural networks. *Economic Modelling, 54*, 40–53.

Nademi, A., & Nademi, Y. (2018). Forecasting crude oil prices by a semiparametric Markov switching model: OPEC, WTI, and Brent cases. *Energy Economics, 74*, 757–766.

Nonejad, N. (2021). Predicting equity premium by conditioning on macroeconomic variables: A prediction selection strategy using the price of crude oil. *Finance Research Letters, 41*, 101792.

Norouzi, N., & Fani, M. (2020). Black gold falls, black plague arise - An OPEC crude oil price forecast using a gray prediction model. *Upstream Oil and Gas Technology, 5*, 100015.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 31.

Rabbi, M. F., Moon, M. H., Dhonno, F. T., Sultana, A., & Abedin, M. Z. (2022). Foreign currency exchange rate prediction using long short-term memory, support vector regression and random forest regression. *Financial data analytics: theory and application* (pp. 251–267). Springer.

Rubaszek, M. (2021). Forecasting crude oil prices with DSGE models. *International Journal of Forecasting, 37*(2), 531–546.

Sajid, S. W., Hasan, M., Rabbi, M. F., & Abedin, M. Z. (2023). An Ensemble LGBM (light gradient boosting machine) approach for Crude oil price prediction. *Novel financial applications of machine learning and deep learning: Algorithms, product modelling, and applications* (pp. 153–165). Springer.

Shajalal, M., Hajek, P., & Abedin, M. Z. (2023). Product backorder prediction using deep neural network on imbalanced data. *International Journal of Production Research, 61*(1), 302–319.

Shin, H., Hou, T., Park, K., Park, C.-K., & Choi, S. (2013). Prediction of movement direction in crude oil prices based on semi-supervised learning. *Decision Support Systems, 55*, 348–358.

Sun, S., Sun, Y., Wang, S., & Wei, Y. (2018). Interval decomposition ensemble approach for crude oil price forecasting. *Energy Economics, 76*, 274–287.

Sun, W., Chen, H., Liu, F., & Wang, Y. (2022). Point and interval prediction of crude oil futures prices based on chaos theory and multiobjective slime mold algorithm. *Annals of Operations Research*. https://doi.org/10.1007/s10479-022-04781-6

Vo, A. H., Nguyen, T., & Le, T. (2020). Brent oil price prediction using Bi-LSTM network. *Intelligent Automation and Soft Computing, 26*(6), 1307–1317.

Wang, J., Athanasopoulos, G., Hyndman, R. J., & Wang, S. (2018). Crude oil price forecasting based on internet concern using an extreme learning machine. *International Journal of Forecasting, 34*(4), 665–677.

Wang, J., Niu, T., Du, P., & Yang, W. (2020). Ensemble probabilistic prediction approach for modeling uncertainty in crude oil price. *Applied Soft Computing, 95*, 106509.

Wang, S., Nan, B., Rosset, S., & Zhu, J. (2011). Random lasso. *The Annals of Applied Statistics, 5*(1), 468.

Wen, F., Gong, X., & Cai, S. (2016). Forecasting the volatility of crude oil futures using HAR-type models with structural breaks. *Energy Economics, 59*, 400–413.

Wu, B., Wang, L., Lv, S. X., & Zeng, Y. R. (2021a). Effective crude oil price forecasting using new text-based and big-data-driven model. *Measurement, 168*, 108468.

Wu, T., Zhang, W., Jiao, X., Guo, W., & Hamoud, Y. A. (2021b). Evaluation of stacking and blending ensemble learning methods for estimating daily reference evapotranspiration. *Computers and Electronics in Agriculture, 184*, 106039.

Xie, W., Yu, L., Xu, S., & Wang, S. (2006). A new method for crude oil price forecasting based on support vector machines. *International conference on computational science* (pp. 444–451). Springer.

Yang, Y., Guo, J. E., Sun, S., & Li, Y. (2021). Forecasting crude oil price with a new hybrid approach and multi-source data. *Engineering Applications of Artificial Intelligence, 101*, 104217.

Yu, L., Dai, W., & Tang, L. (2016). A novel decomposition ensemble model with extended extreme learning machine for crude oil price forecasting. *Engineering Applications of Artificial Intelligence, 47*, 110–121.

Yuan, J., Li, J., & Hao, J. (2023). A dynamic clustering ensemble learning approach for crude oil price forecasting. *Engineering Applications of Artificial Intelligence, 123*, 106408.

Zhang, Y., Wahab, M. I. M., & Wang, Y. (2022). Forecasting crude oil market volatility using variable selection and common factor. *International Journal of Forecasting, 39*, 486–502.

Zhao, Y., Li, J., & Yu, L. (2017). A deep learning ensemble approach for crude oil price forecasting. *Energy Economics, 66*, 9–16.

Zhao, Y., Zhang, W., Gong, X., & Wang, C. (2021). A novel method for online real-time forecasting of crude oil price. *Applied Energy, 303*, 117588.