A CNN–LSTM model for gold price time-series forecasting

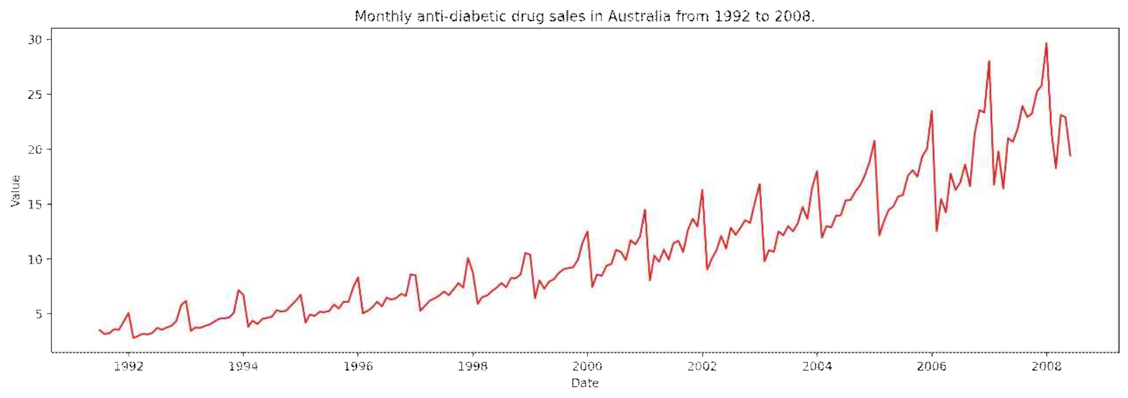
Songyi Lee
June, 21, 2021
Dongguk University

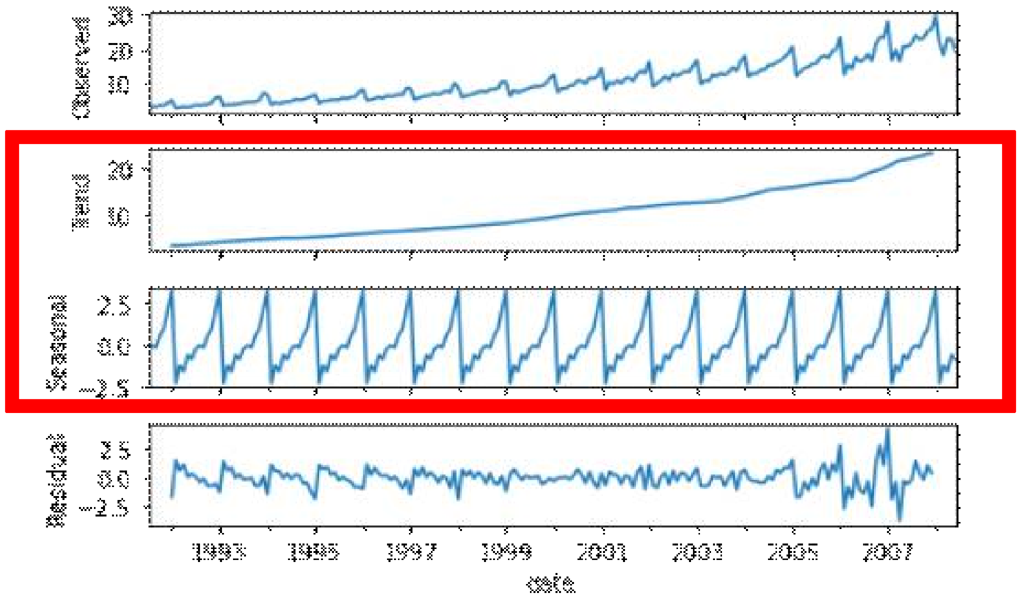
BACKGROUND 1

Time Series Data

- Time series data are data that is collected at different points in time.
- Everything we are interested in to make money is time series data
- The time series data are all about stocks, oil prices, bond interest rates, natural resources prices, house prices etc.

Example : Monthly anti-diabetic drug sales





- Divided the observed data(sale data) by the trend, seasonal and residual.
- Trend and seasonal are constant (able to predict the sales in 2009 only through the dividing.)
- But, Gold, Silver and oil prices do not have a clear trend or seasonality.

Gold Precise forecast

- The gold price constitutes a significant part of the economy of banks and stock markets.
- Significance emerged after the 2008 financial crisis. (2008 : $800 per ounce)
- It soared to $1,800 at the end of 2011 (Prices remain high)
- Gold have expanded to alternative investments.

Gold Precise forecast

- Fluctuations in gold prices could result in huge profits.
- But, also have increased the risk of investment
- Precise forecast in financial area will facilitate the development of economy.

Gold Price Determinants

- Dollar (Substitute)
- Inflation
- COVID-19
- Interest rate
- safe asset preference
- Other demands

- Trend of gold price is impacted by numerous factors.

Dataset

Table 1 Descriptive statistics for gold daily prices

Statistic	Value
Minimum	100.50
Mean	118.4784
Maximum	133.10
Median	119.325
SD	6.8776
Skewness	– 0.5509
Kurtosis	2.6991

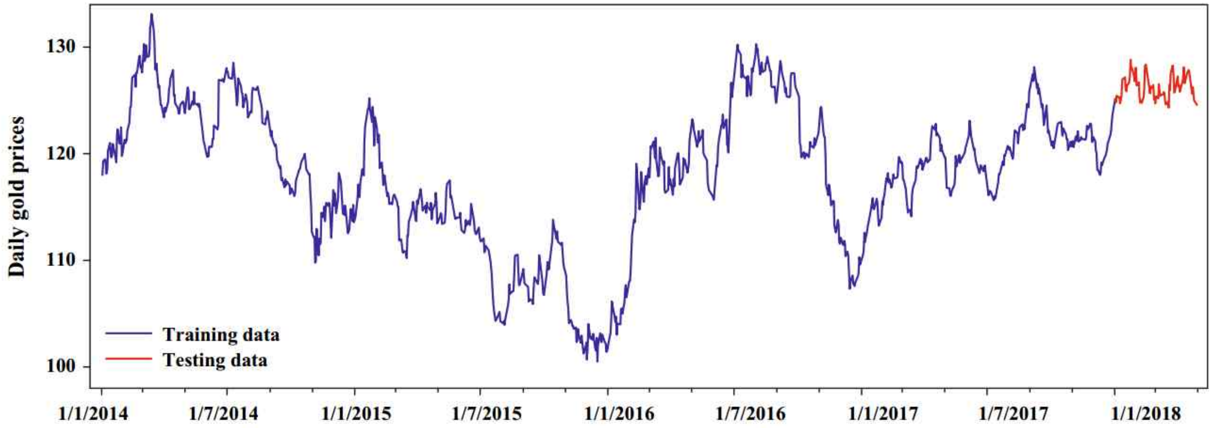


Fig. 3 Daily gold price trend from January 2014 to April 2018

- Daily gold prices in USD : Jan 2014 ~ Apr 2018.
- Descriptive statistics: Minimum, mean, maximum, median, standard deviation (SD), skewness, kurtosis.
- Data scaling: Transformed utilizing a natural logarithm (ln)
- Source :Yahoo (http://finance.yahoo.com)

PRIOR STUDIES

- Statistical Model

✓ ARIMA (But only deal with the linear relationships)

- Machine Learning Model

✓ SVM

✓ DT

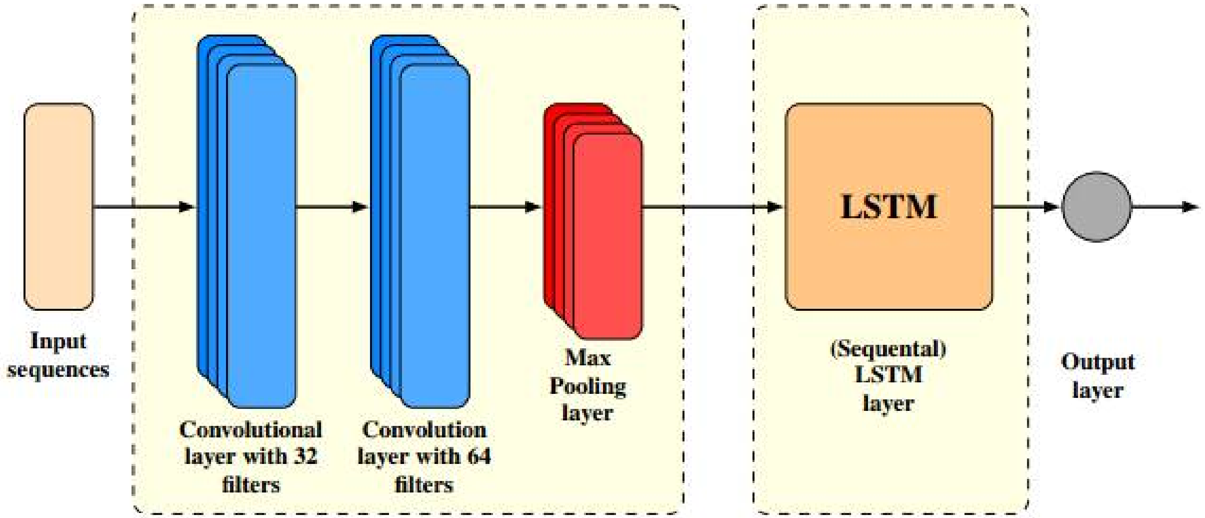
- Deep Learning Model

✓ LSTM

✓ CNN

✓ MLP

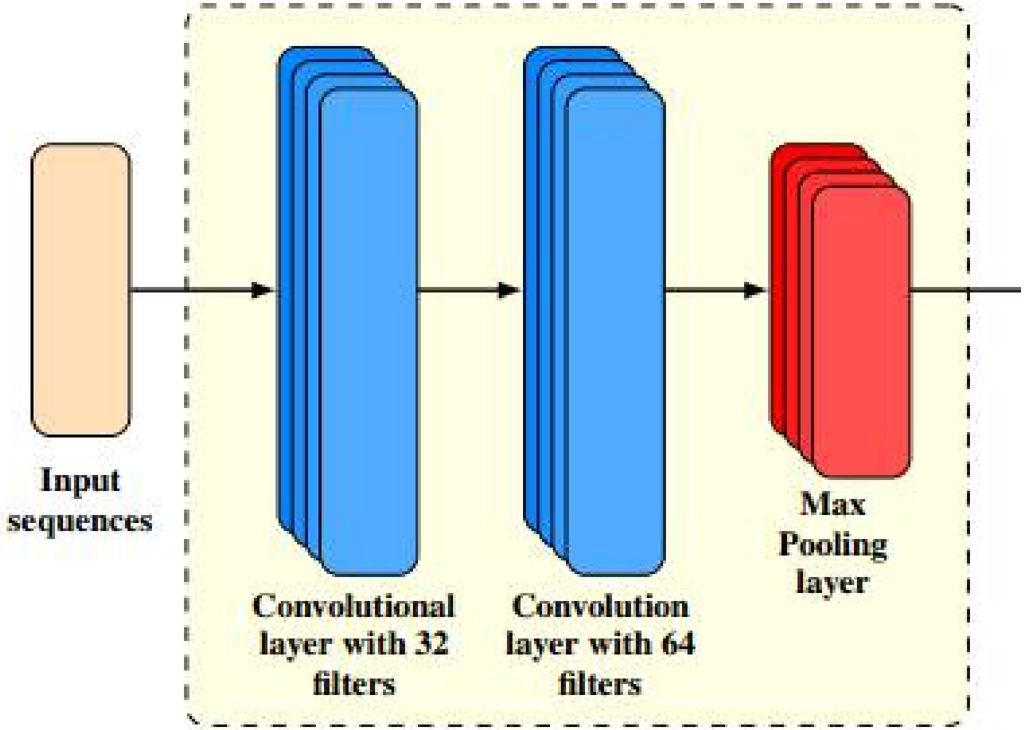
PROPOSED MODEL :CNN-LSTM MODEL



1. Convolutional layers : extract useful knowledge and learn the internal representation of time-series data.
2. LSTM: effective for identifying short-term and long-term dependencies.

- Proposed model is to efficiently combine the advantages of these deep leaning techniques.

CNN



- CNN allows to extract peripheral features between variables.

LSTM

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i),$$

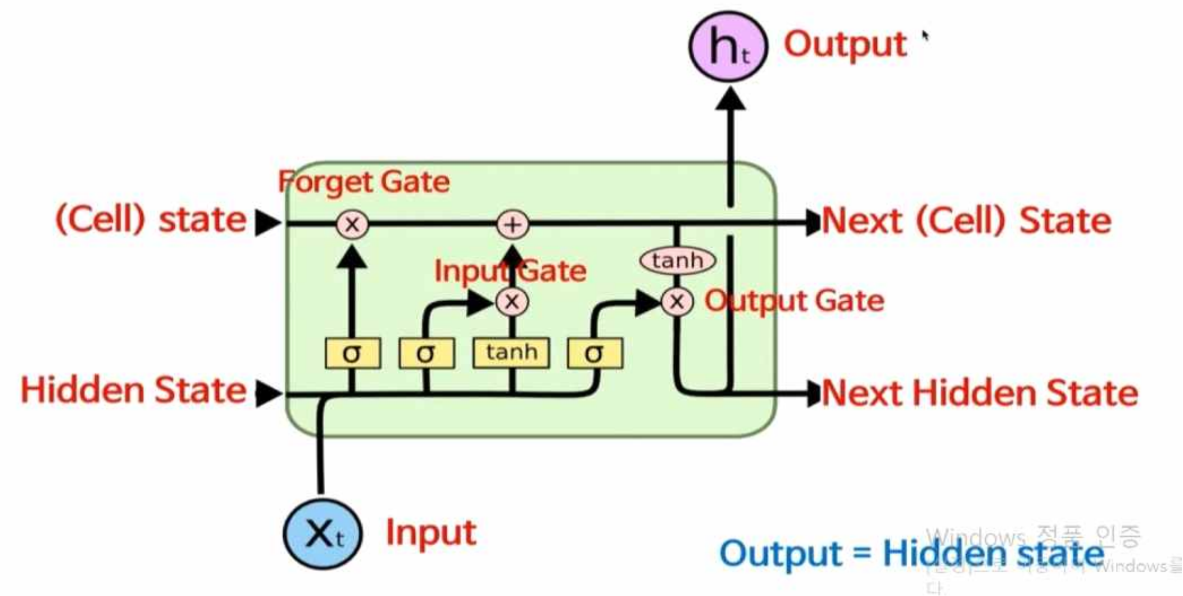
$$f_t = \sigma(U_g x_t + W_g h_{t-1} + b_g),$$

$$c_t^* = \tanh(U_c x_t + W_c h_{t-1} + b_c),$$

$$c_t = g_t \odot c_{t-1} + i_t \odot c_t^*,$$

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o),$$

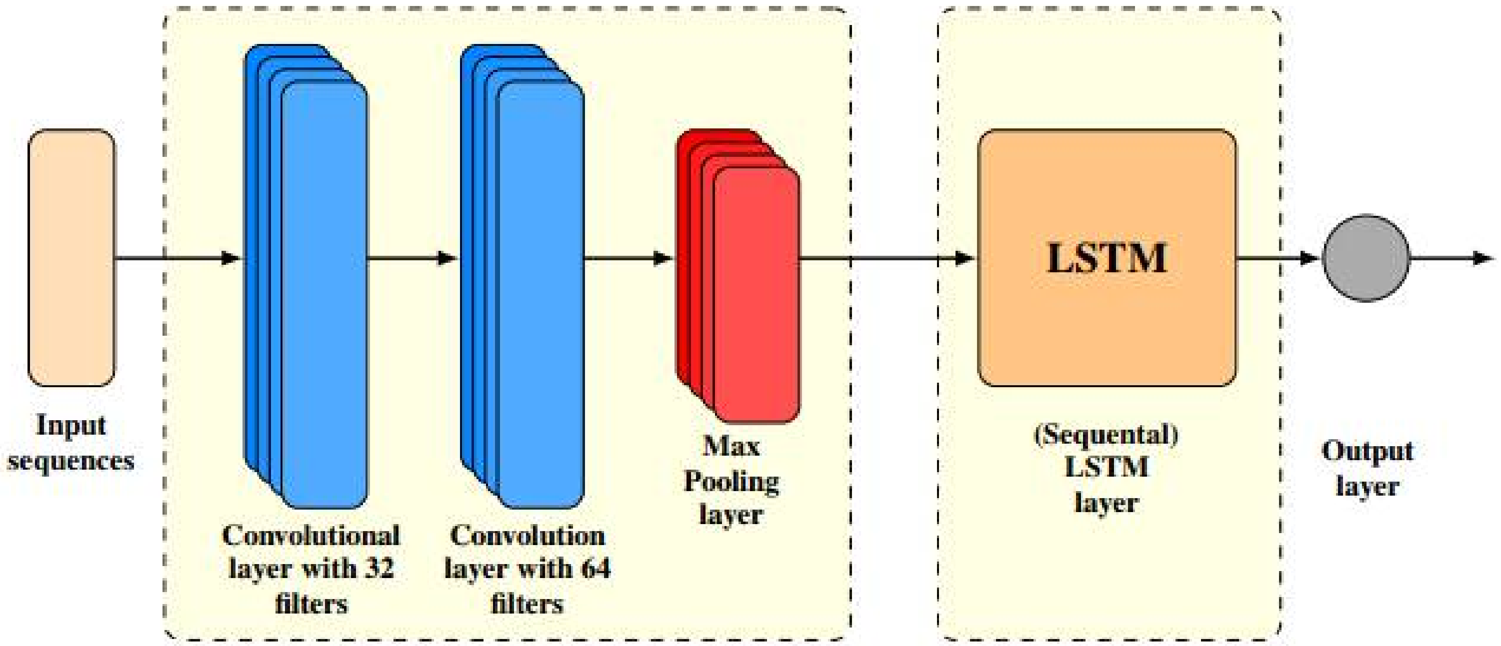
$$h_t = o_t \odot \tanh(c_t).$$



- LSTM is a type of RNNs which learns long-term dependencies through the utilization of feedback connections.
- RNNs suffer from the vanishing gradient problem which restricts the model to learn long-range dependencies.
- LSTM come to solve this problem by storing useful information on memory cells and vanishing unnecessary information.

CNN-LSTM MODEL 1

Fig. 1 Proposed CNN–LSTM$_1$ model architecture with two convolutional layers, a pooling layer, a LSTM layer and an output layer



CNN-LSTM MODEL 2

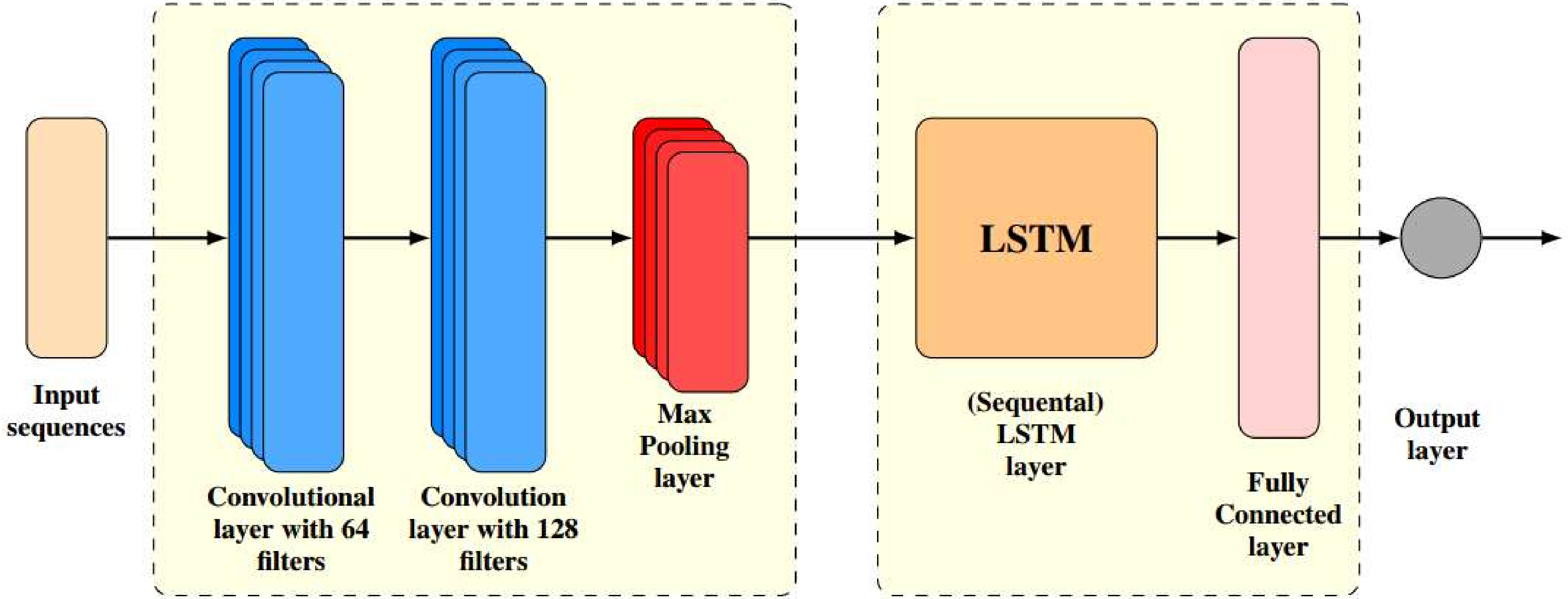


Fig. 2 Proposed CNN–$LSTM_2$ model architecture with two convolutional layers, a pooling layer, a LSTM layer, a fully connected layer and an output layer

Table 2 Parameter specification of all forecasting models

Model	Description
SVR	Kernel = RBF, $C = 1.0$, Tolerance $= 10^{-3}$
FFNN	1 hidden layer with 3, 3 and 5 neurons for $F = 4, 6$ and 9, respectively
$LSTM_1$	LSTM layer with 100 units
$LSTM_2$	LSTM layer with 200 units
$LSTM_3$	LSTM layer with 100 units
	LSTM layer with 50 units
$LSTM_4$	LSTM layer with 100 units
	LSTM layer with 100 units
	Fully connected layer with 32 neurons
CNN–$LSTM_1$	Convolutional layer with 32 filters of size (2,)
	Convolutional layer with 64 filters of size (2,)
	Max pooling layer with size (2,)
	LSTM layer with 100 units
CNN–$LSTM_2$	Convolutional layer with 64 filters of size (2,)
	Convolutional layer with 128 filters of size (2,)
	Max pooling layer with size (2,)
	LSTM layer with 200 units
	Fully connected layer with 32 neurons

EXPERIMENTS

Table 3 Performance comparison based on MAE, RMSE, accuracy, AUC, sensitivity and specificity of the proposed CNN–LSTM models against traditional regression models for forecasting horizon equal to 4

Model	MAE	RMSE	ACC (%)	AUC	SEN	SPE
SVR	0.0567	0.0554	48.68	0.476	0.316	0.658
FFNN	0.0149	0.0190	47.37	0.489	0.421	0.526
$LSTM_1$	0.0222	0.0272	52.63	0.526	0.500	0.553
$LSTM_2$	0.0136	0.0172	50.00	0.500	0.500	0.500
$LSTM_3$	0.0242	0.0236	53.55	0.538	0.553	0.526
$LSTM_4$	0.0128	0.0162	51.32	0.513	0.579	0.447
CNN–$LSTM_1$	0.0099	0.0127	55.26	0.553	0.553	0.553
CNN–$LSTM_2$	0.0079	0.0082	51.58	0.519	0.553	0.519

Table 4 Performance comparison based on MAE, RMSE, accuracy, AUC, sensitivity and specificity of the proposed CNN–LSTM models against traditional regression models for forecasting horizon equal to 6

Model	MAE	RMSE	ACC (%)	AUC	SEN	SPE
SVR	0.0571	0.0576	48.68	0.476	0.316	0.658
FFNN	0.0242	0.0255	50.00	0.501	0.474	0.526
$LSTM_1$	0.0205	0.0259	52.63	0.526	0.468	0.584
$LSTM_2$	0.0148	0.0184	50.19	0.522	0.495	0.549
$LSTM_3$	0.0251	0.0246	53.95	0.540	0.526	0.553
$LSTM_4$	0.0138	0.0174	52.63	0.526	0.500	0.553
CNN–$LSTM_1$	0.0094	0.0110	56.81	0.577	0.577	0.558
CNN–$LSTM_2$	0.0082	0.0095	55.53	0.555	0.579	0.500

EXPERIMENTS

Table 5 Performance comparison based on MAE, RMSE, accuracy, AUC, sensitivity and specificity of the proposed CNN–LSTM models against traditional regression models for forecasting horizon equal to 9

Model	MAE	RMSE	ACC (%)	AUC	SEN	SPE
SVR	0.0571	0.0579	48.68	0.476	0.316	0.658
FFNN	0.0287	0.0301	46.05	0.461	0.368	0.553
$LSTM_1$	0.0200	0.0254	52.37	0.524	0.440	0.608
$LSTM_2$	0.0158	0.0196	51.89	0.509	0.326	0.732
$LSTM_3$	0.0194	0.0243	52.63	0.526	0.474	0.579
$LSTM_4$	0.0141	0.0178	51.62	0.508	0.400	0.679
CNN–$LSTM_1$	0.0095	0.0117	55.26	0.553	0.500	0.618
CNN–$LSTM_2$	0.0089	0.0100	54.21	0.542	0.474	0.632

Table 6 Confusion matrices of $LSTM_3$, $LSTM_4$, CNN–$LSTM_1$ and CNN–$LSTM_2$ for forecasting horizon equal to 4

	Down	Up		Down	Up
Up	20	18	Down	17	21
Down	17	21	Up	16	22
$LSTM_3$			$LSTM_4$		
	Down	Up		Down	Up
Down	21	17	Down	19	19
Up	17	21	Up	17	21
CNN–$LSTM_1$			CNN–$LSTM_2$		

EXPERIMENTS

Table 7 Confusion matrices of $LSTM_3$, $LSTM_4$, CNN–$LSTM_1$ and CNN–$LSTM_2$ for forecasting horizon equal to 6

	Down	Up		Down	Up
Down	21	17	Down	19	19
Up	18	20	Up	17	21
$LSTM_3$			$LSTM_4$		

	Down	Up		Down	Up
Down	22	16	Down	23	15
Up	15	23	Up	19	19
CNN–$LSTM_1$			CNN–$LSTM_2$		

Table 8 Confusion matrices of $LSTM_3$, $LSTM_4$, CNN–$LSTM_1$ and CNN–$LSTM_2$ for forecasting horizon equal to 9

	Down	Up		Down	Up
Down	22	16	Down	24	14
Up	20	18	Up	24	14
$LSTM_3$			$LSTM_4$		

	Down	Up		Down	Up
Down	24	14	Down	23	15
Up	19	19	Up	20	18
CNN–$LSTM_1$			CNN–$LSTM_2$		

CONCLUSION

1. Proposed a new forecasting model CNN-LSTM for the prediction of gold price.

2. First model reported the best forecasting performance for regression problems (MAE, RMSE)

3. Second model exhibited the best performance for the classification problem of predicting the gold movement

4. With additional convolutional layers provides a significant boost in increasing the forecasting performance

5. Possible that additional optimized configuration and mostly feature engineering could further improve the forecasting ability

THANK YOU