FULL LEGAL NAME	LOCATION (COUNTRY)	EMAIL ADDRESS	MARK X FOR ANY NON-CONTRIBUTING MEMBER
Marin Stoyanov	Bulgaria	azonealerts@gmx.com	
Brian Luna Patino	United Kingdom	brian1311@hotmail.co.uk	
Long Chen	China	chenlongpku@163.com	

Statement of integrity: By typing the names of all group members in the text boxes below, you confirm that the assignment submitted is original work produced by the group (excluding any non-contributing members identified with an "X" above).	
Team member 1	Marin Stoyanov
Team member 2	Brian Luna Patino
Team member 3	Long Chen

Use the box below to explain any attempts to reach out to a non-contributing member. Type (N/A) if all members contributed. **Note:** You may be required to provide proof of your outreach to non-contributing members upon request.
N/A

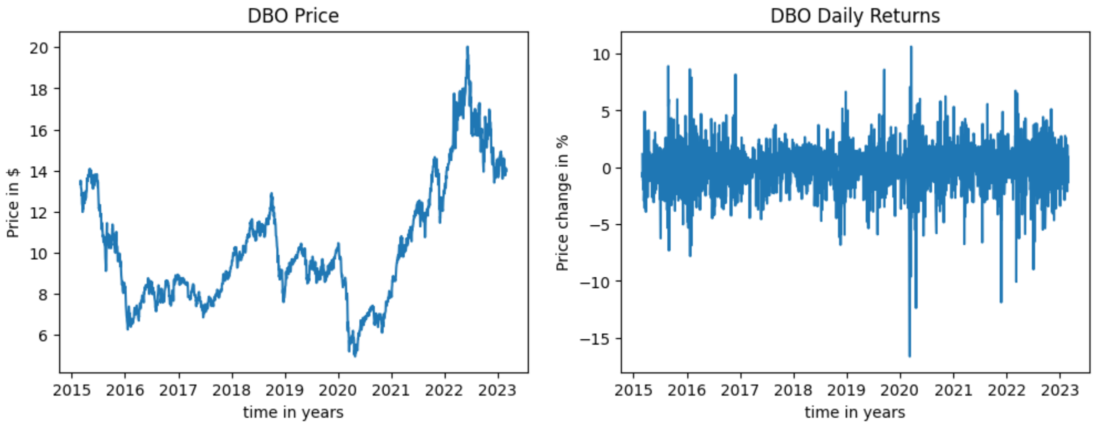
Step 1: Single Train/Test Split

1.a Collecting Data

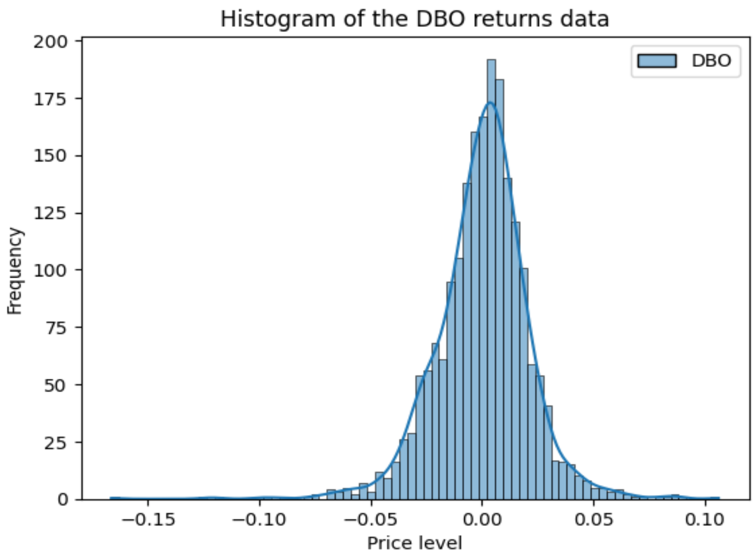
In a time of high turbulence when the hot events that are happening around the world like Russia vs Ukraine war [1], Israel vs Hamas[2], or the Houthis attacking ships in the Red Sea and Gulf of Aden [3], we are very interested in predicting the oil price going up or down in the near future since the oil is a very important part of today's economy.

The Invesco DB Oil Fund is an ETF incorporated in the USA, which is a "rules-based index composed of futures contracts on light sweet crude oil" [4].

We download data from Yahoo Finance spanning from March, 2015 to March, 2023.



We draw the distribution of the daily returns, and we can see it is centered on 0 but left-skewed.



1.b Problem Formulation

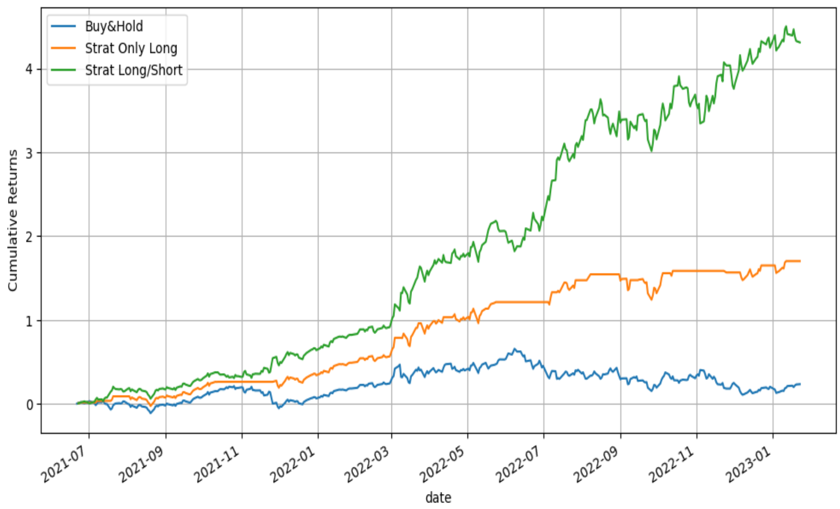
The task is to predict whether the 25-day future return is positive or not, using a 30-day window of daily returns or closing prices. The dataset contains 1,958 samples after processing. The labels are slightly imbalanced.

```
dbo_data.label.value_counts()
```

```
1    1121
0     867
```

We wonder if it is advantageous to trade DBO ETF if we know its 25-day future returns.

```
Strat Return Long Only = 170.22244059588127 %
Strat Return = 430.96770359068273 %
Buy and Hold Return = 23.502762369918372 %
```



Trading Strategies Based on Ground-Truth 25-Day Future Returns

Both Long and Long/Short strategies are very profitable compared to Buy&Hold strategy, if we can train a model that can predict the 25-day future returns accurately.

Therefore, we formulate the problem as a binary classification problem to predict whether the 25-day future return is positive (label 1) or negative (label 0).

To construct training and test sets with or without information leakage, we define a helper function:

```Python
def create_train_test(df, features, label, window_size, test_size, verbose=False,           input_scale=True, leakage_band=0):
      '''
      leakage_band controls how many samples at the end of the
      training data will be discarded to prevent information leak
      '''
```

```
    split_index = int(df.shape[0] - test_size)
    X_train_set = df[features].iloc[:split_index - leakage_band]
    y_train_set = df[label].iloc[:split_index - leakage_band]
    X_test_set = df[features].iloc[split_index - window_size:]
    y_test_set = df[label].iloc[split_index - window_size:]
    ...
```

The test set would always contain `test_size` samples. The size of the training set depends on `leakage_band`, which controls how many samples prior to the test set will be excluded from the training set.

When `leakage_band` is set to 0, then there is clear information leakages, since the beginning of the test set would be very similar to the end of the training set.

When `leakage_band` is set to at least `window_size`, the information leakage is mitigated.

1.c Training Models

We train the models using batch_size of 64 and Adam optimizer with a learning rate of 1e-3. Each model is trained for 100 epochs with early stopping (patience 20) using a validating set that is 15% of the training set.

The model structures and evaluations are listed below. We find that the LSTM model is considerably harder (taking longer epochs) to train (without having better generalization necessarily).

MLP

Model: "sequential"

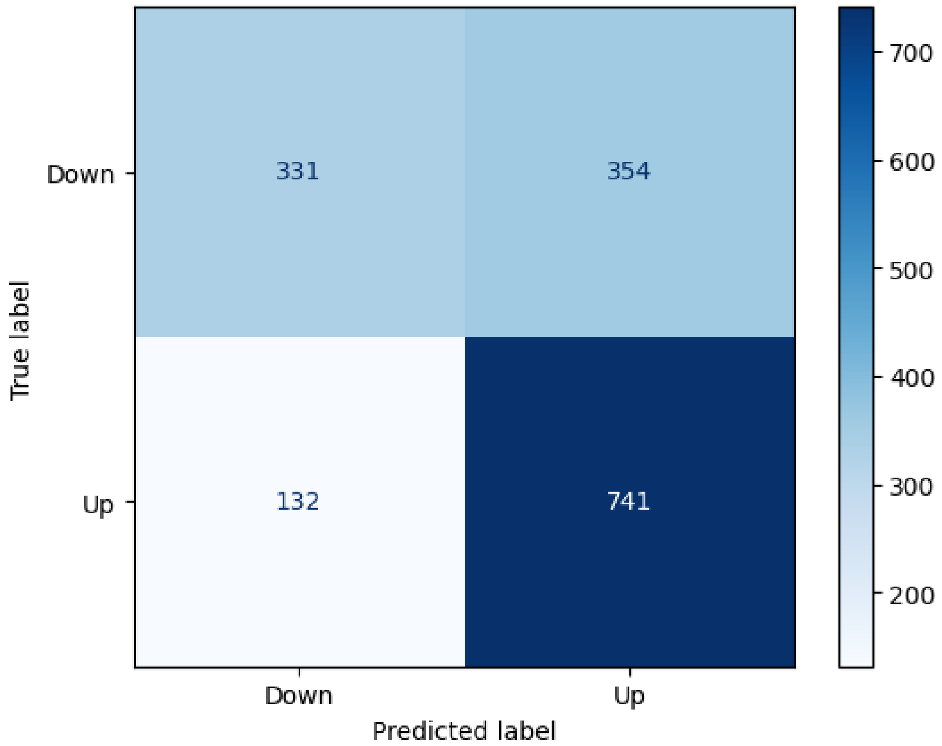
Layer (type)	Output Shape	Param #
dense (Dense)	(None, 50)	1550
dense_1 (Dense)	(None, 50)	2550
dense_2 (Dense)	(None, 30)	1530
dense_3 (Dense)	(None, 10)	310
dense_4 (Dense)	(None, 1)	11

Total params: 5951 (23.25 KB)
Trainable params: 5951 (23.25 KB)
Non-trainable params: 0 (0.00 Byte)

Model Structure of MLP

```
eval_classification(X_train, y_train, model_mlp)
```

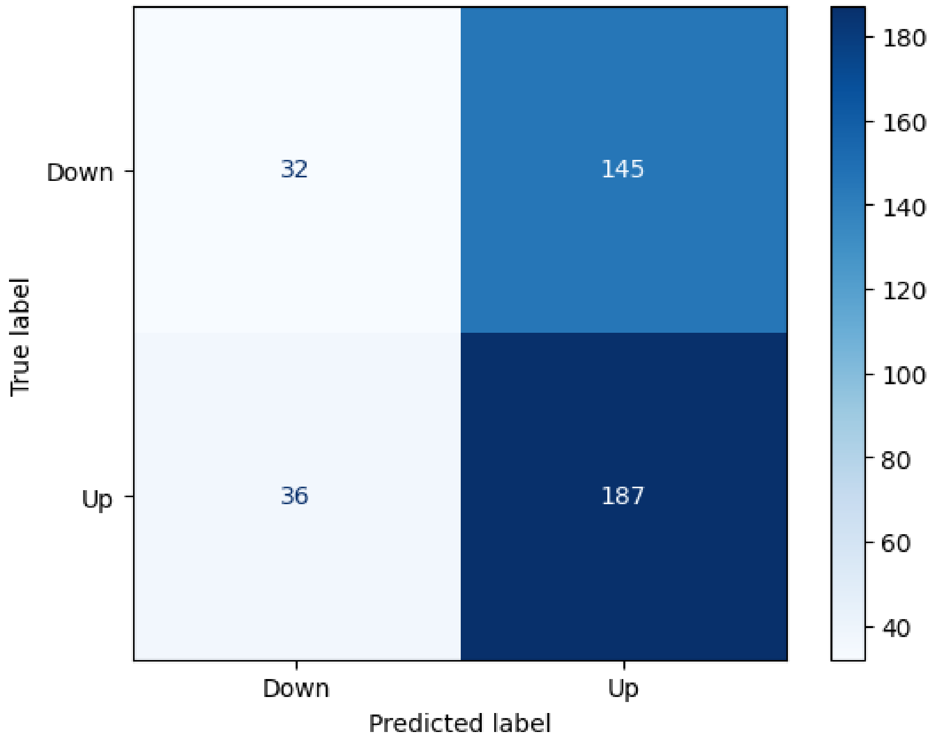
```
49/49 [==============================] - 0s 1ms/step - loss: 0.5702 - accuracy: 0.6881
Accuracy over test: 68.81%
49/49 [==============================] - 0s 984us/step
              precision    recall  f1-score   support

           0       0.71      0.48      0.58       685
           1       0.68      0.85      0.75       873

    accuracy                           0.69      1558
   macro avg       0.70      0.67      0.66      1558
weighted avg       0.69      0.69      0.68      1558
```



MLP Performance on the Training Set

```
eval_classification(X_test, y_test, model_mlp)
```

```
13/13 [==============================] - 0s 1ms/step - loss: 0.7484 - accuracy: 0.5475
Accuracy over test: 54.75%
13/13 [==============================] - 0s 1ms/step
              precision    recall  f1-score   support

           0       0.47      0.18      0.26       177
           1       0.56      0.84      0.67       223

    accuracy                           0.55       400
   macro avg       0.52      0.51      0.47       400
weighted avg       0.52      0.55      0.49       400
```



MLP Performance on the Test Set

LSTM

Model: "sequential"

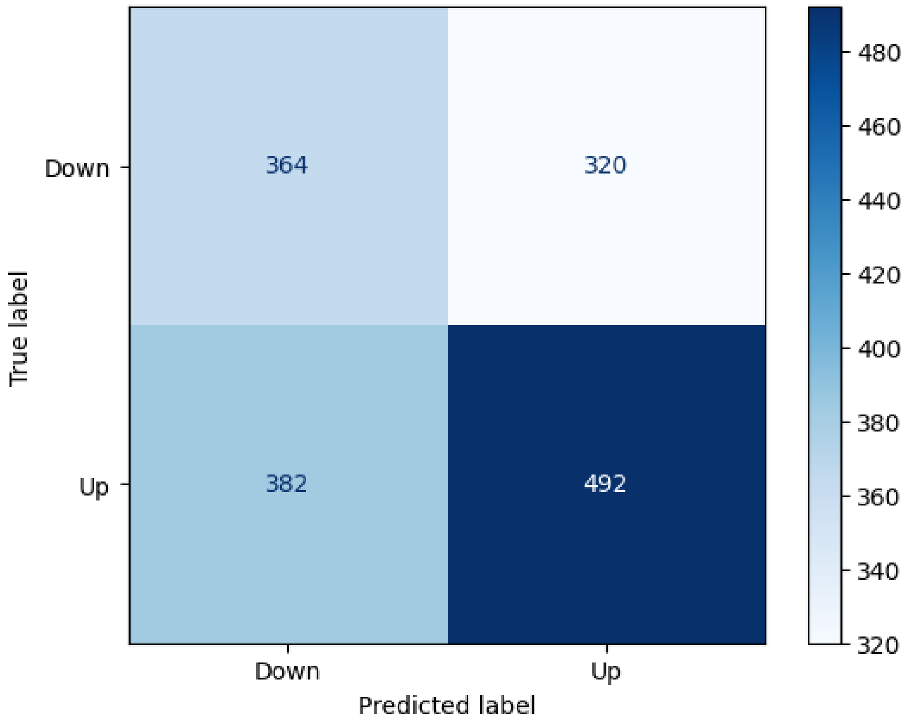
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 50)	10400
lstm_1 (LSTM)	(None, 50)	20200
dense (Dense)	(None, 30)	1530
dense_1 (Dense)	(None, 10)	310
dense_2 (Dense)	(None, 1)	11

Total params: 32451 (126.76 KB)
Trainable params: 32451 (126.76 KB)
Non-trainable params: 0 (0.00 Byte)

Model Structure of LSTM

```
eval_classification(X_train, y_train, model_lstm)
```

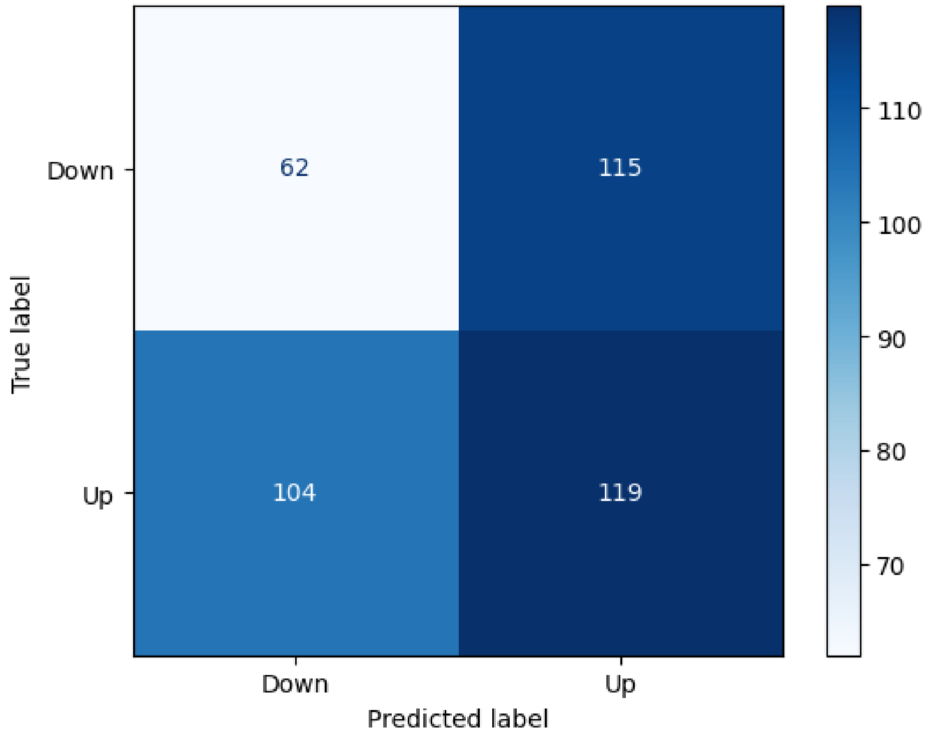
```
49/49 [==============================] - 1s 11ms/step - loss: 0.6654 - accuracy: 0.5494
Accuracy over test: 54.94%
49/49 [==============================] - 2s 13ms/step
              precision    recall  f1-score   support

           0       0.49      0.53      0.51       684
           1       0.61      0.56      0.58       874

    accuracy                           0.55      1558
   macro avg       0.55      0.55      0.55      1558
weighted avg       0.55      0.55      0.55      1558
```



LSTM Performance on the Training Set

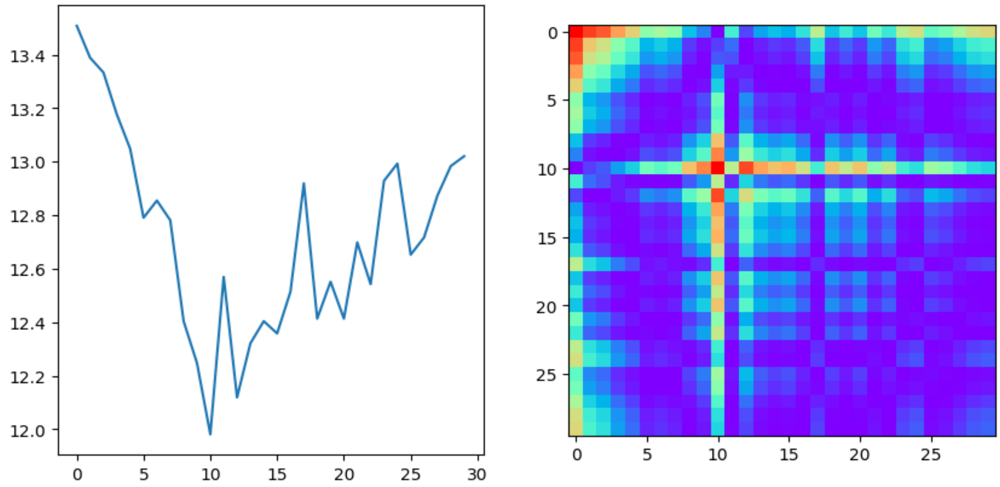
```
eval_classification(X_test, y_test, model_lstm)
```

```
13/13 [==============================] - 0s 11ms/step - loss: 0.7325 - accuracy: 0.4525
Accuracy over test: 45.25%
13/13 [==============================] - 0s 11ms/step
              precision    recall  f1-score   support

           0       0.37      0.35      0.36       177
           1       0.51      0.53      0.52       223

    accuracy                           0.45       400
   macro avg       0.44      0.44      0.44       400
weighted avg       0.45      0.45      0.45       400
```



LSTM Performance on the Test Set

CNN on GAF



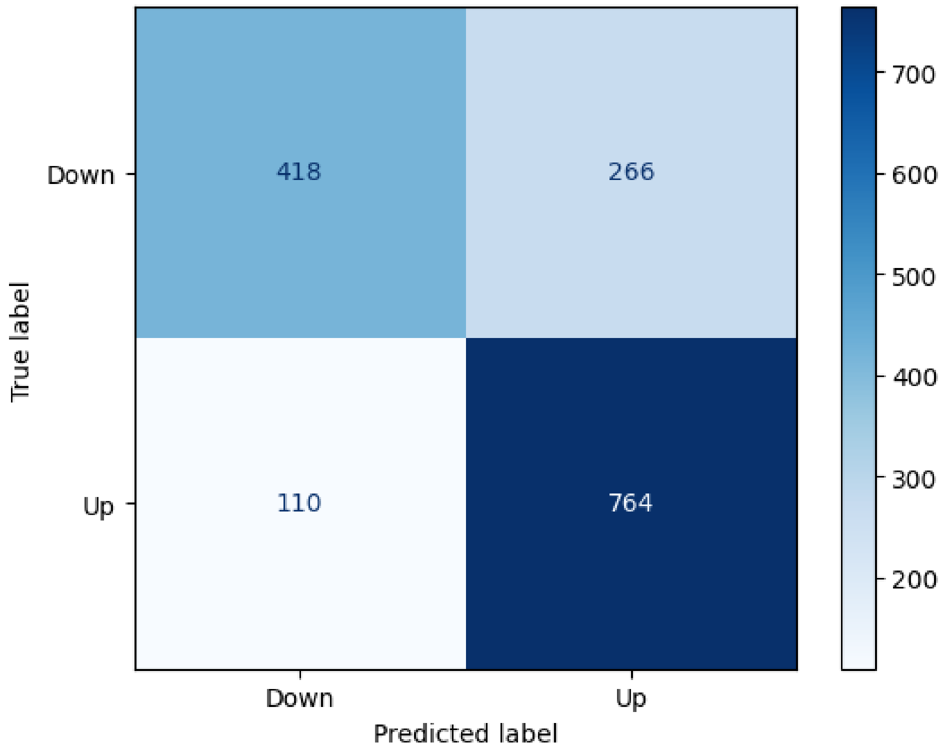
Example of a Window of Closing Prices and its Corresponding GAF

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 28, 28, 16)        160

 max_pooling2d (MaxPooling2  (None, 14, 14, 16)        0
 D)

 conv2d_1 (Conv2D)           (None, 12, 12, 32)        4640

 max_pooling2d_1 (MaxPoolin  (None, 6, 6, 32)          0
 g2D)

 conv2d_2 (Conv2D)           (None, 4, 4, 64)          18496

 max_pooling2d_2 (MaxPoolin  (None, 2, 2, 64)          0
 g2D)

 flatten (Flatten)           (None, 256)               0

 dense (Dense)               (None, 64)                16448

 dropout (Dropout)           (None, 64)                0

 dense_1 (Dense)             (None, 1)                 65

=================================================================
Total params: 39809 (155.50 KB)
Trainable params: 39809 (155.50 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Model Structure of CNN

```
eval_classification(X_train_gaf,  y_train,  model_cnn)
```

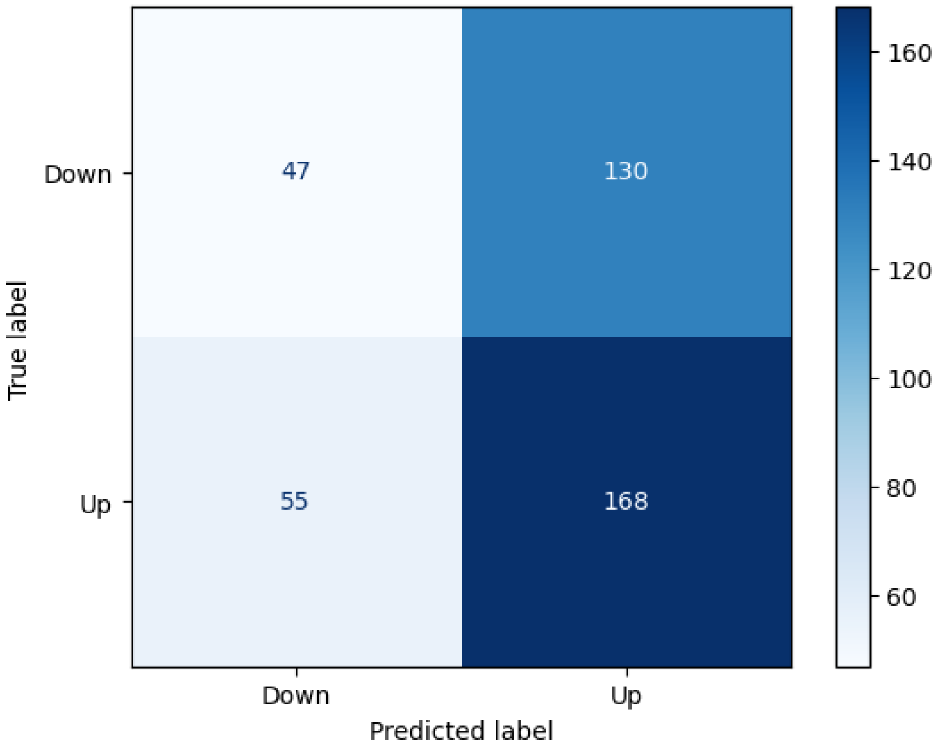
```
49/49 [==============================] - 0s 6ms/step - loss: 0.5212 - accuracy: 0.7587
Accuracy over test: 75.87%
49/49 [==============================] - 0s 6ms/step
              precision    recall  f1-score   support

           0       0.79      0.61      0.69       684
           1       0.74      0.87      0.80       874

    accuracy                           0.76      1558
   macro avg       0.77      0.74      0.75      1558
weighted avg       0.76      0.76      0.75      1558
```



CNN Performance on the Training Set

```
eval_classification(X_test_gaf, y_test, model_cnn)
```

```
13/13 [==============================] - 0s 8ms/step - loss: 0.7391 - accuracy: 0.5375
Accuracy over test: 53.75%
13/13 [==============================] - 0s 6ms/step
              precision    recall  f1-score   support

           0       0.46      0.27      0.34       177
           1       0.56      0.75      0.64       223

    accuracy                           0.54       400
   macro avg       0.51      0.51      0.49       400
weighted avg       0.52      0.54      0.51       400
```



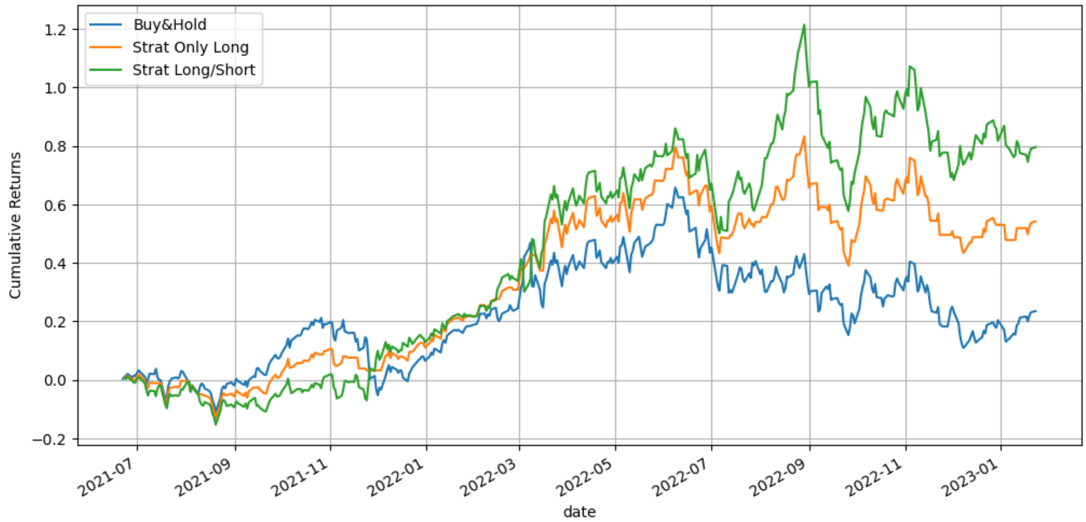
CNN Performance on the Test Set

1.d Backtest

The test period is from 2021-06-22 to 2023-01-23.

MLP

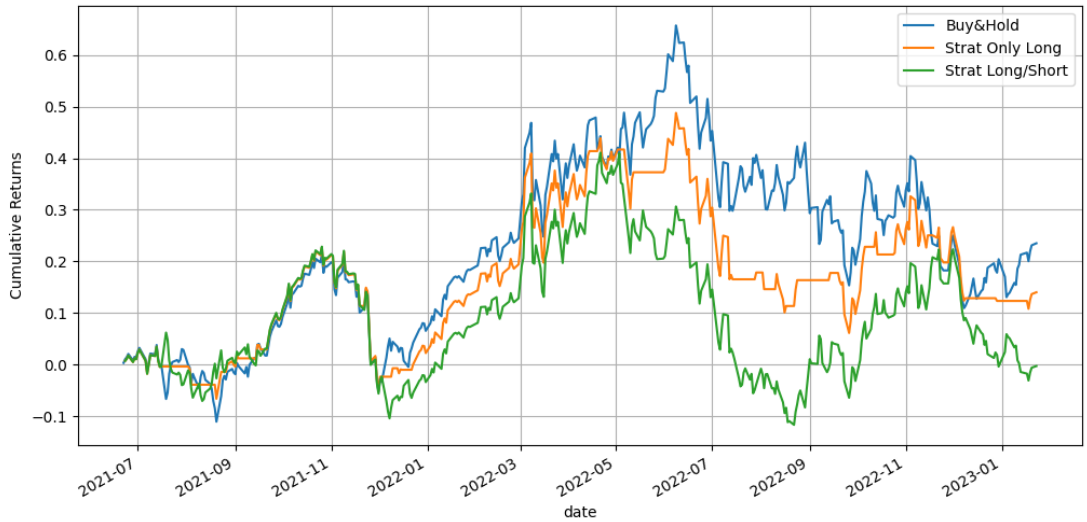
```
Strat Return Long Only = 54.15602496599414 %
Strat Return = 79.58534955646648 %
Buy and Hold Return = 23.502762369918507 %
```



MLP Backtest Result during the Test Period

LSTM

```
Strat Return Long Only = 14.004872094482511 %
Strat Return = -0.30069829232378664 %
Buy and Hold Return = 23.502762369918507 %
```

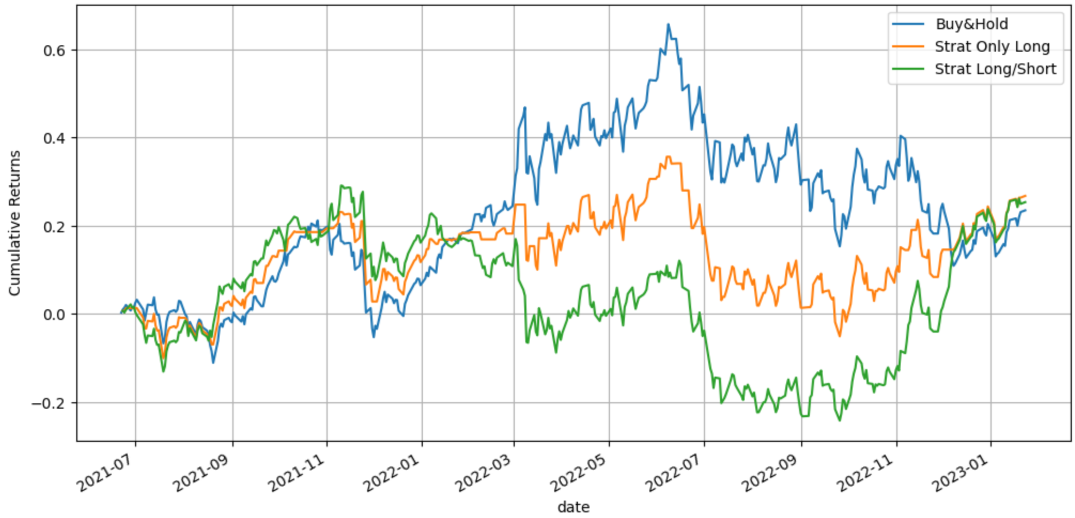


LSTM Backtest Result during the Test Period

CNN

```
backtest(df_res)
```

```
Strat Return Long Only = 26.783804030805534 %
Strat Return = 25.361335945855636 %
Buy and Hold Return = 23.502762369918507 %
```



CNN Backtest Result during the Test Period

Discussion

- All three models perform similarly to the ground-truth backtest result prior to 2021-09, indicating higher accuracy at the beginning of the test priod, which is expected due to information leakage.
- Both the LSTM model and the CNN model have a similar number of trainable parameters, yet the former has lower accuracy and worse cumulative returns compared to the latter.
- Though the MLP model has only 6,000 trainable parameters and the CNN model has almost 40,000 trainable parameters, the former outperforms the latter on the test set, indicating a higher degree of overfitting of the CNN model on the training set even with early-stopping.

Step 2: Non-Anchored Walk Forward Method

In both step 2 and 3, the training strategy is to train the model on the training set for 10 epochs with a learning rate of 1e-3 before continuing to train the model with a learning rate of 1e-4 until early-stopping occurs by loss on the validation set, which is 15% of the training set.

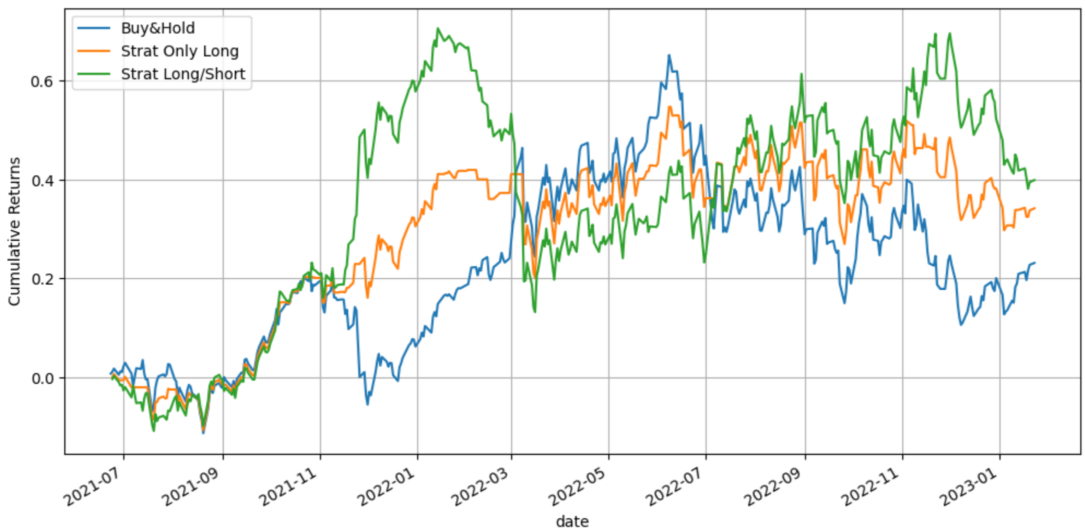
In the future, we would like to try to fine-tune the model instead of re-initializing the model at each walk-forward step.

2.a Train/Test Split 500/500

MLP

```
Strat Return Long Only = 133.403806725102 %
Strat Return = 144.12204869908604 %
Buy and Hold Return = 81.279545776673 %
```

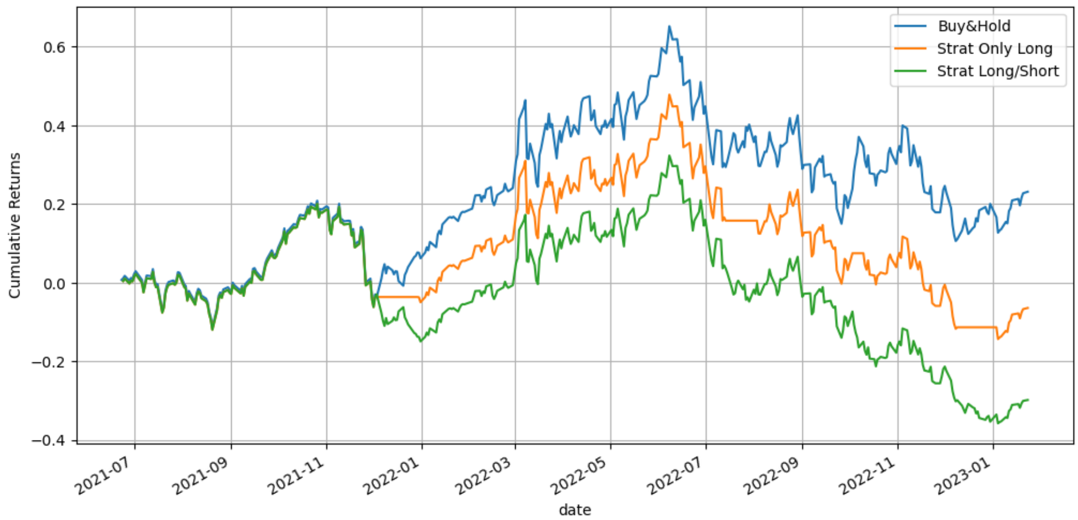

```
Strat Return Long Only = 34.171042163671814 %
Strat Return = 39.89174729890721 %
Buy and Hold Return = 23.11560369565777 %
```



LSTM

Strat Return Long Only = 3.013209336235012 %
Strat Return = -53.37911191972382 %
Buy and Hold Return = 81.279545776673 %

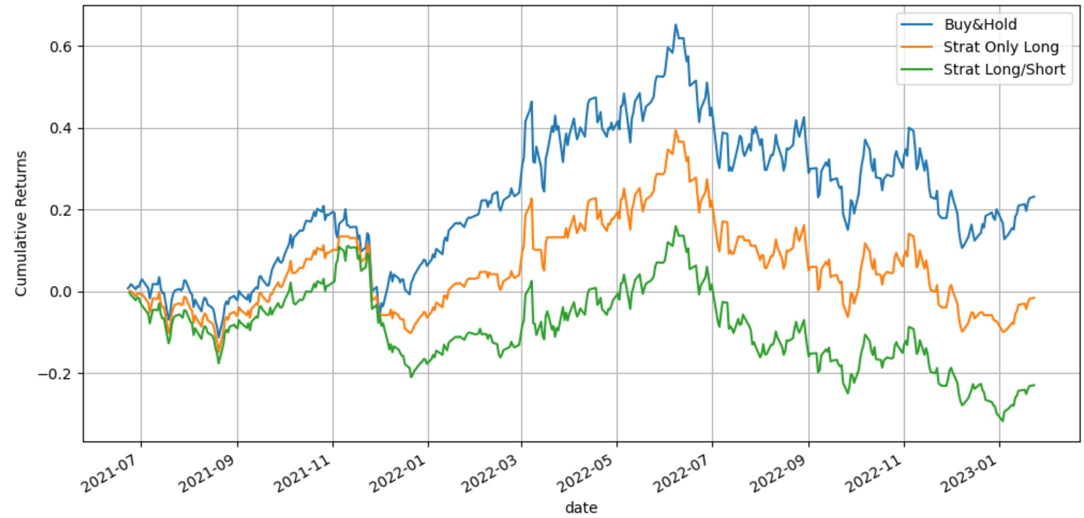
Strat Return Long Only = -6.4396296742577075 %
Strat Return = -29.844703331958613 %
Buy and Hold Return = 23.11560369565777 %



CNN

Strat Return Long Only = 14.30702659605707 %
Strat Return = -42.4306643737538 %
Buy and Hold Return = 81.279545776673 %

Strat Return Long Only = -1.5939367825589468 %
Strat Return = -22.986556443249196 %
Buy and Hold Return = 23.11560369565777 %

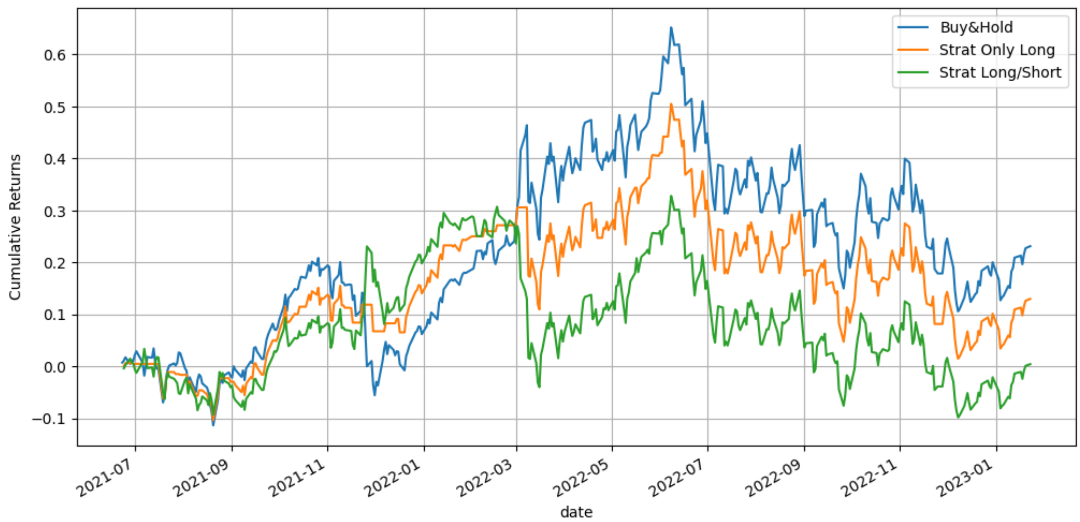


2.b Train/Test Split 500/100

MLP

```
Strat Return Long Only = 16.057731706475685 %
Strat Return = -36.866125051567366 %
Buy and Hold Return = 81.279545776673 %
```

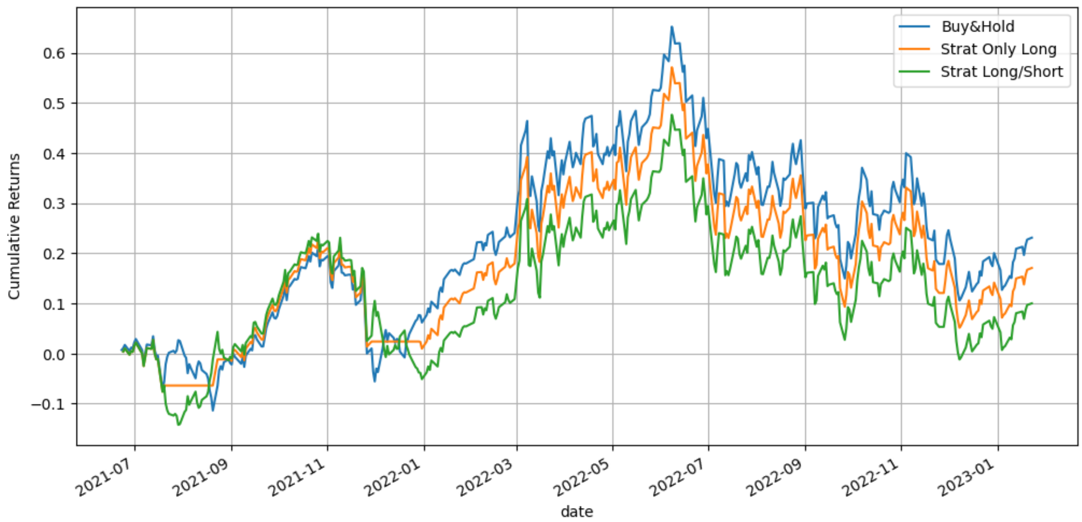

```
Strat Return Long Only = 12.96717066732549 %
Strat Return = 0.405763598717912 %
Buy and Hold Return = 23.11560369565777 %
```



LSTM

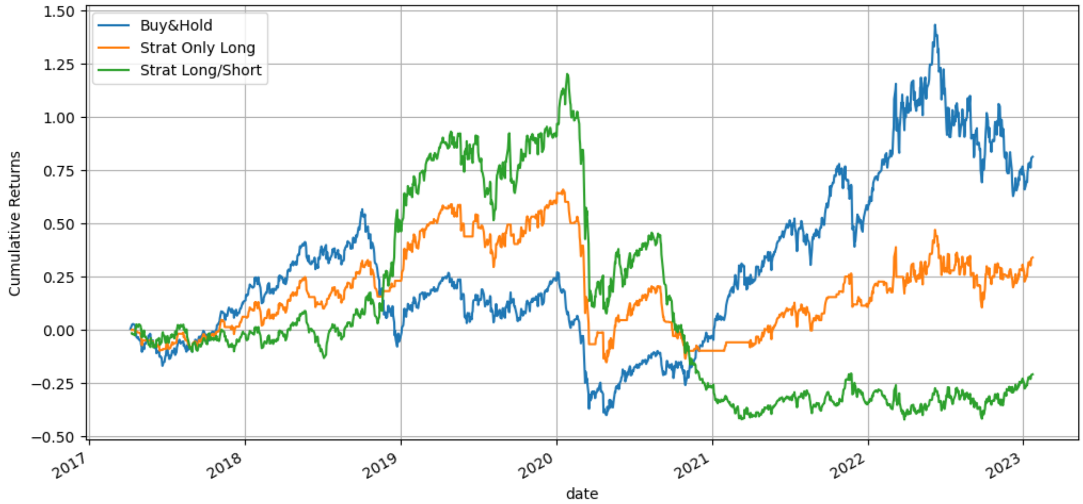
Strat Return Long Only = 16.98387976362603 %
Strat Return = -34.98495352840742 %
Buy and Hold Return = 81.279545776673 %

Strat Return Long Only = 17.074209917444193 %
Strat Return = 10.019260523160757 %
Buy and Hold Return = 23.11560369565777 %

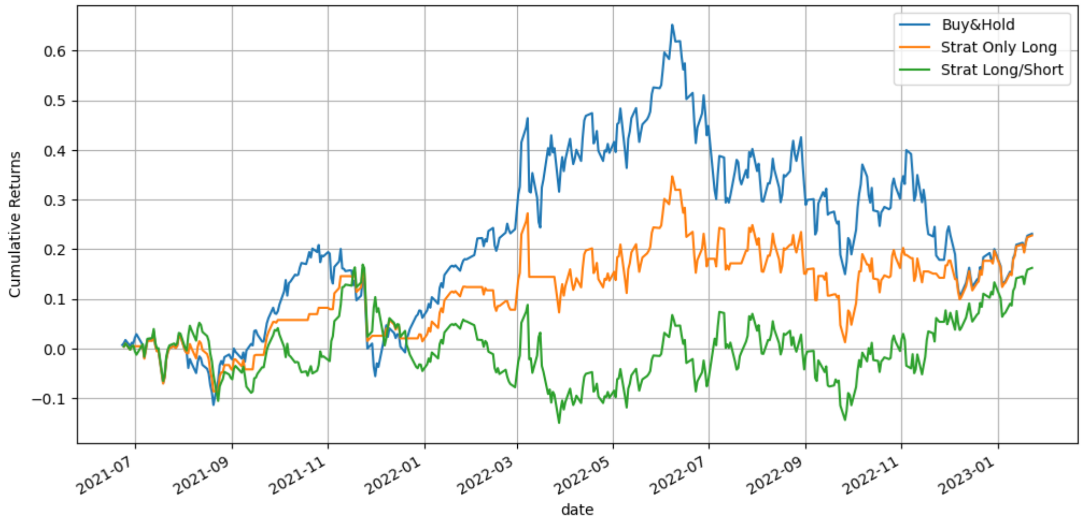


CNN

Strat Return Long Only = 34.01587378238753 %
Strat Return = -20.83591196947916 %
Buy and Hold Return = 81.279545776673 %



Strat Return Long Only = 22.785097440867762 %
Strat Return = 16.2491013477839 %
Buy and Hold Return = 23.11560369565777 %



2.c Compare to Step 1

We define a common test period as the test period in Step 1, which is from 2021-06-22 to 2023-01-23.

The difference is that in this setup, three models are trained in total during walk-forward, and the last two models are used to predict the labels in this common test period. We observe a slightly worse performance in terms of backtest, possibly to the less accurate predictions at the beginning of the common test period.

2.d Compare 2.a and 2.b

The trained epochs and the performances of the models are more volatile in step 2.b than in step 2.a.

Step 3: Non-Anchored Walk Forward Method without Information Leakage

3.a Reduce the Extent of Leakage

As discussed in 1.b, we use the le parameter `leakage_band` in the `create_train_test` to control how many samples prior to the test set will be excluded from the training set.

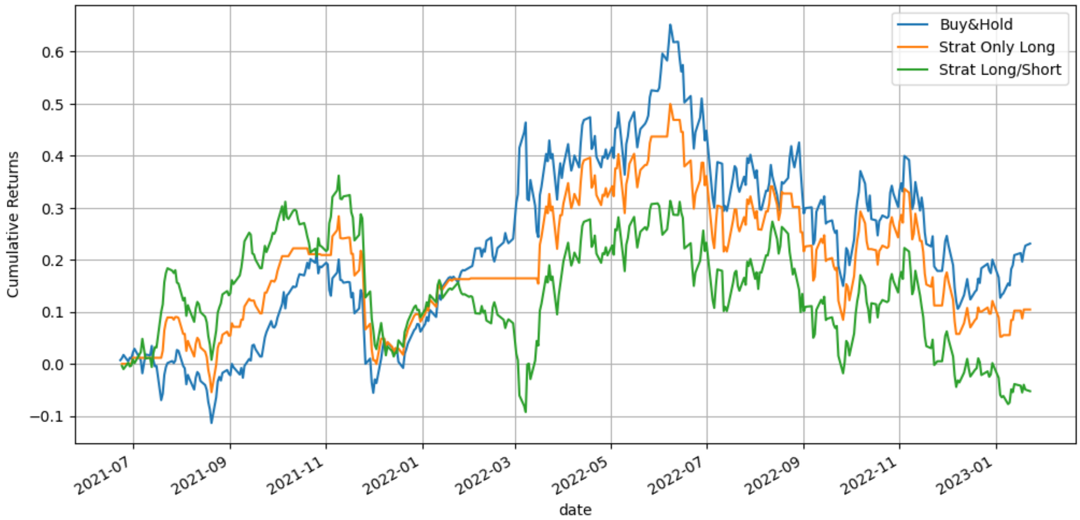
Here, to reduce the extent of information leakage, we set `leakage_band` as `window_size * 2` in the following experiments.

3.b Train/Test Split 500/500

MLP

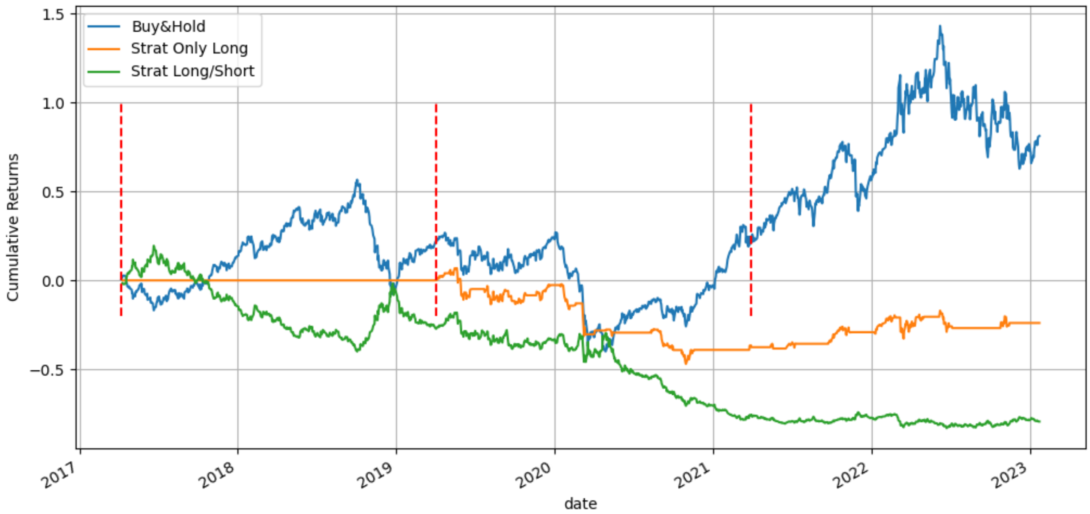
```
Strat Return Long Only = 53.811111878378945 %
Strat Return = 9.951795976237921 %
Buy and Hold Return = 81.279545776673 %
```

Strat Return Long Only = 10.43917028492729 %
Strat Return = -5.251541411650362 %
Buy and Hold Return = 23.11560369565777 %

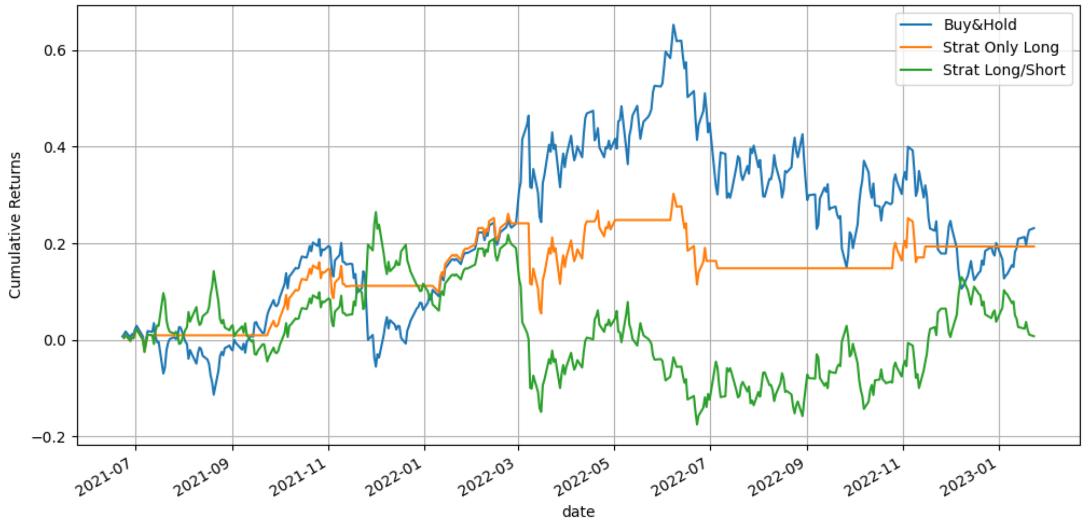


LSTM

Strat Return Long Only = -23.972107671849574 %
Strat Return = -79.45383220879243 %
Buy and Hold Return = 81.279545776673 %



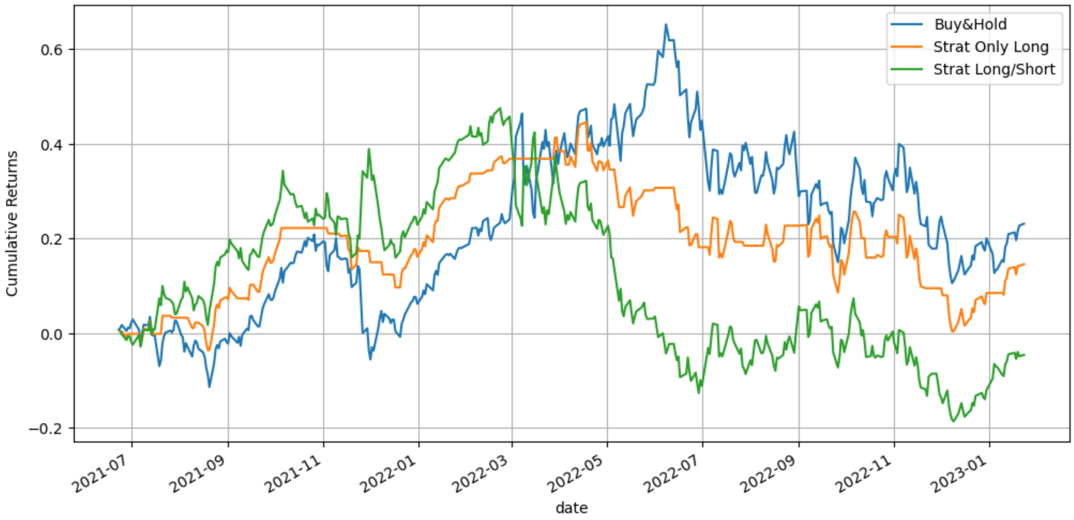
Strat Return Long Only = 19.318948682170635 %
Strat Return = 0.7503713008462976 %
Buy and Hold Return = 23.11560369565777 %



CNN

Strat Return Long Only = 99.55246637504835 %
Strat Return = 54.97693835803019 %
Buy and Hold Return = 81.279545776673 %

Strat Return Long Only = 14.565442055356147 %
Strat Return = -4.591046081501105 %
Buy and Hold Return = 23.11560369565777 %

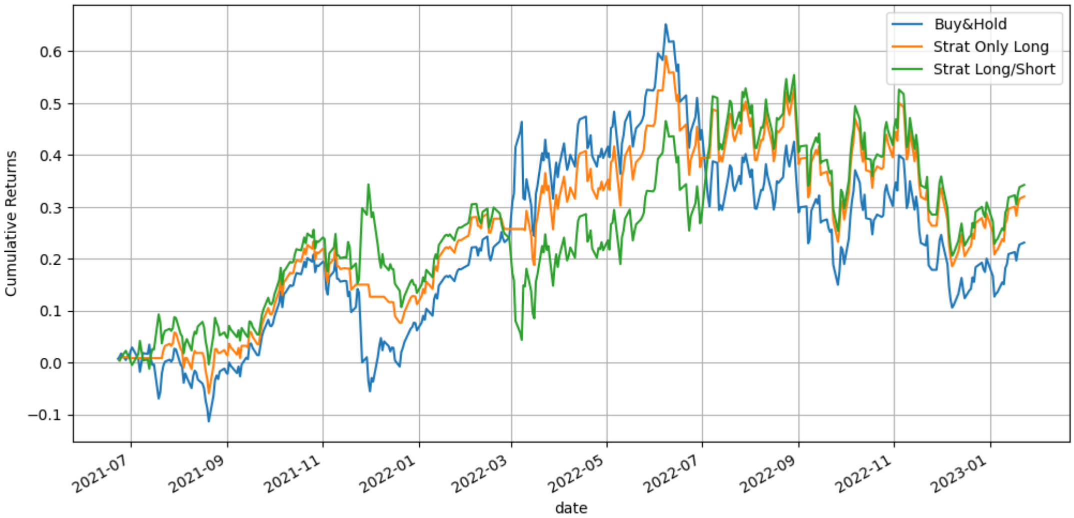


3.c Train/Test Split 500/100

MLP

Strat Return Long Only = 45.08894114906133 %
Strat Return = -1.0897681823201166 %
Buy and Hold Return = 81.279545776673 %

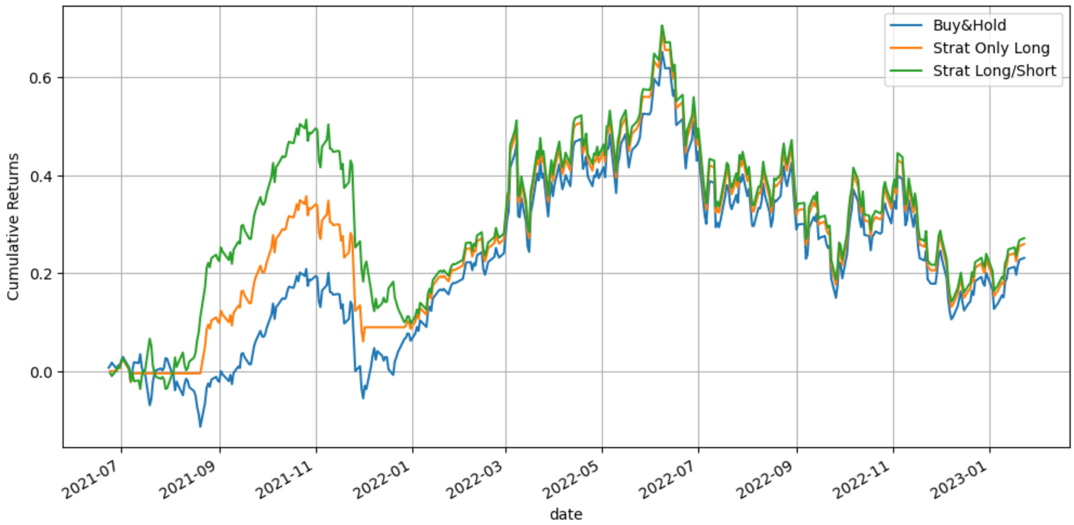
Strat Return Long Only = 31.99426687718374 %
Strat Return = 34.23641326691431 %
Buy and Hold Return = 23.11560369565777 %



LSTM

Strat Return Long Only = 70.9359788311393 %
Strat Return = 35.799148563123936 %
Buy and Hold Return = 81.279545776673 %

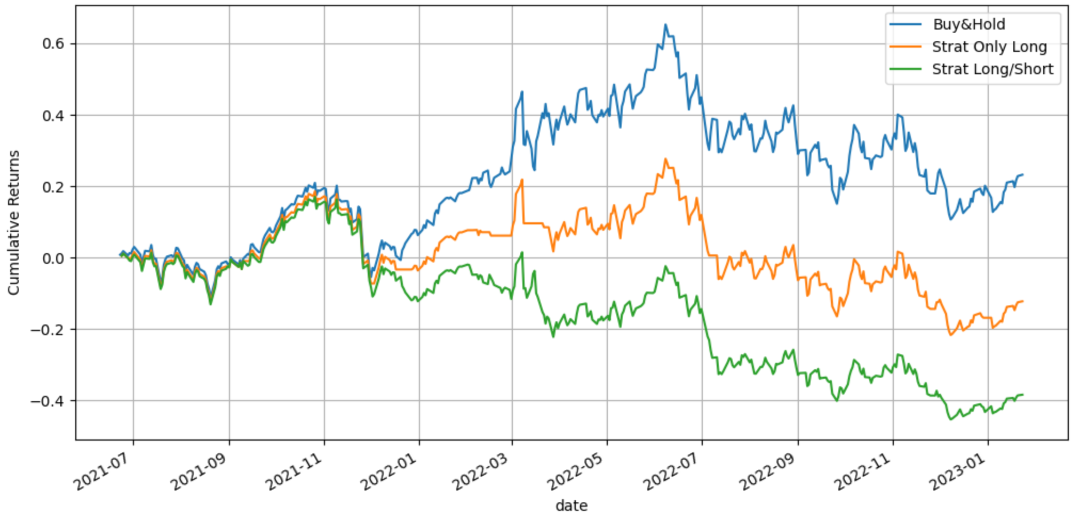
Strat Return Long Only = 25.940388731075092 %
Strat Return = 27.14168932719967 %
Buy and Hold Return = 23.11560369565777 %



CNN

Strat Return Long Only = 0.11394260736616868 %
Strat Return = -53.741354685783385 %
Buy and Hold Return = 81.279545776673 %

```
Strat Return Long Only = -12.344188833240377 %
Strat Return = -38.50572332929261 %
Buy and Hold Return = 23.115603695657777 %
```



3.d Compare to Step 2

Model	Test Size	Total Test Accuracy	Total Test C.R. Long	Total Test C.R. Long/Short	Common Test Accuracy	Common Test C.R. Long	Common Test C.R. Long/Short
Step 2 MLP	500	0.52	133.40%	144.12%	0.51	34.17%	39.89%
Step 2 MLP	100	0.52	16.05%	-36.87%	0.52	12.97%	0.41%
Step 2 LSTM	500	0.48	3.01%	-53.38%	0.50	-6.44%	-29.84%
Step 2 LSTM	100	0.53	16.98%	-34.98%	0.47	17.07%	10.02%
Step 2 CNN	500	0.52	14.31%	-42.43%	0.52	-1.6%	-22.99%
Step 2 CNN	100	0.55	34.02%	-20.84%	0.51	22.79%	16.25%
Step 3	500	0.51	53.81%	9.95%	0.53	10.44%	-5.25%

MLP							
Step 3 MLP	100	0.54	45.089	-1.09%	0.53	31.99%	34.24%
Step 3 LSTM	500	0.40	-23.97%	-79.45%	0.48	19.32%	0.75%
Step 3 LSTM	100	0.48	70.93%	35.80%	0.52	25.94%	27.14%
Step 3 CNN	500	0.51	99.55%	54.97%	0.56	14.56%	-4.59%
Step 3 CNN	100	0.52	0.11%	-53.74%	0.49	-12.34%	-38.50%

The backtest results are all worsened (except LSTM)，which can probably attributed to reduced information leakage.

Reference

1. Hanna Arhirova，Jim Heintz; Russia launches sweeping attack on Ukraine’s power sector, a sign of possible escalation, WORLD NEWS Updated 1:02 PM GMT+2, March 23, 2024; https://apnews.com/article/ukraine-russia-war-electric-power-attack-90218fcbed4fbf196aa8148a4aae6239
2. Ben Cahill, Energy Market Implications of the Israel-Hamas Conflict, Center for Strategic and International Studies, CSIS, Published October 11, 2023, https://www.csis.org/analysis/energy-market-implications-israel-hamas-conflict
3. Alexandra Sharp, Deadly Houthi Attack Escalates Threat to Red Sea Shipping, WORLD BRIEF March 7, 2024, https://foreignpolicy.com/2024/03/07/houthis-true-confidence-red-sea-shipping-attacks-deaths-casualties/
4. https://www.invesco.com/us/financial-products/etfs/product-detail?audienceType=Investor&ticker=DBO