Annexure of Results

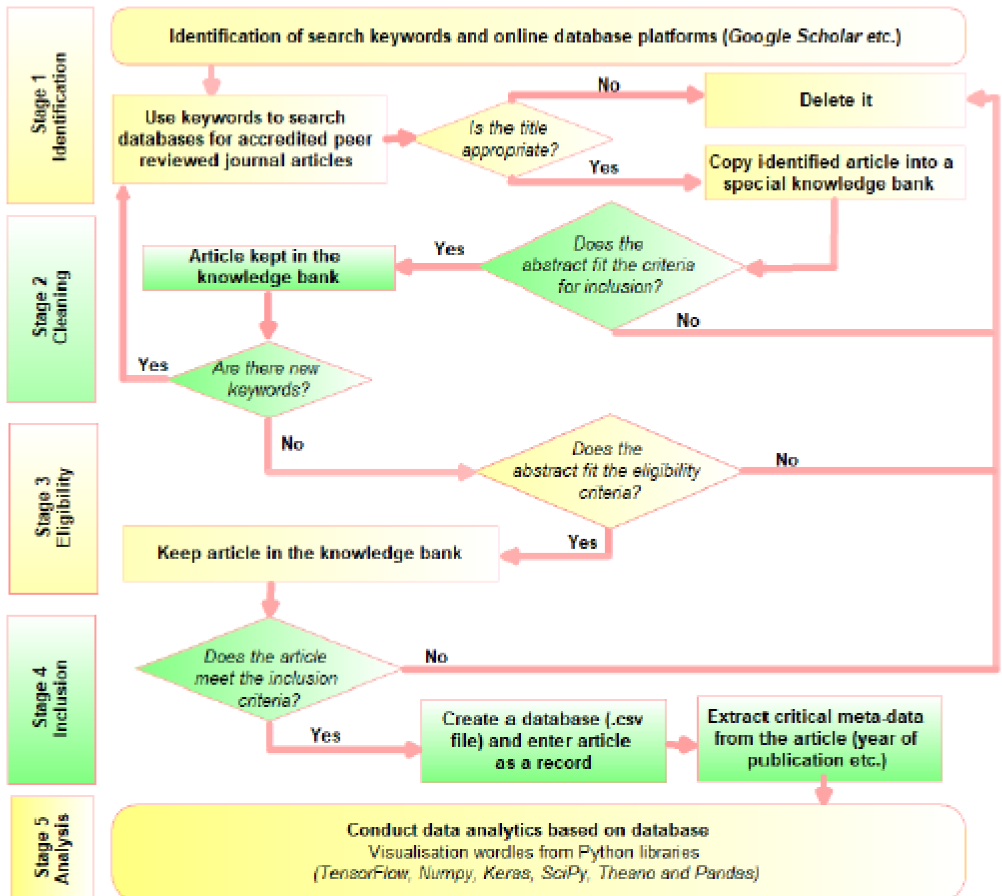

Figure 1: Systematic literature review methodology of Step 1

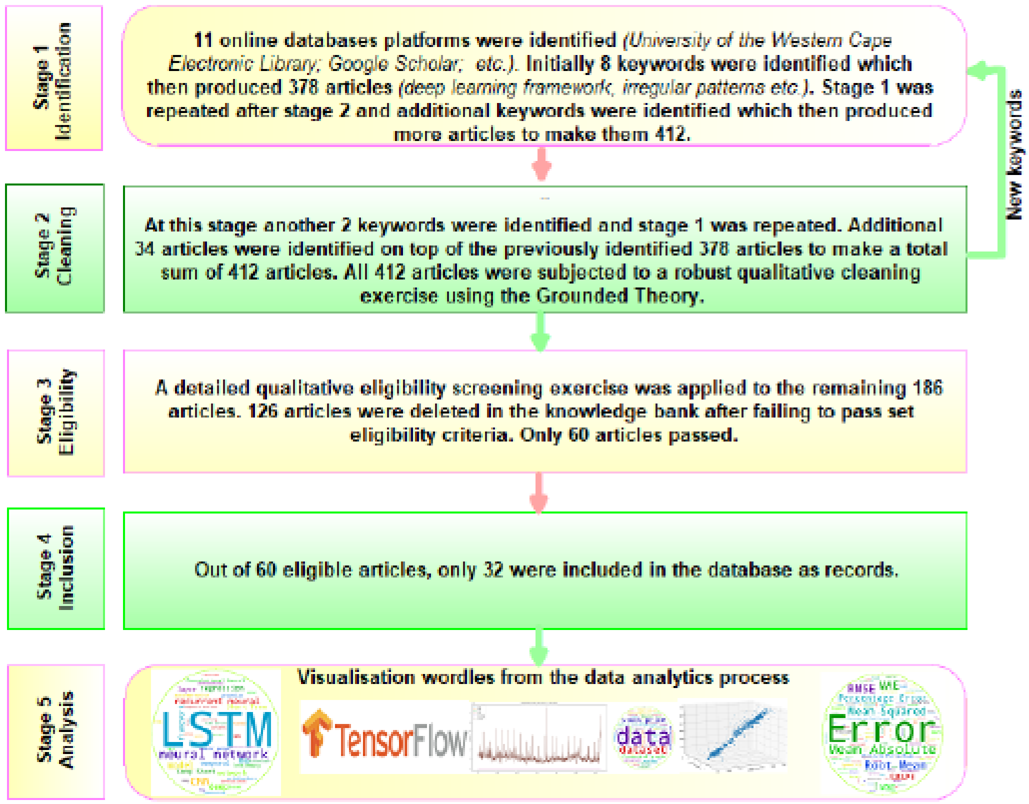

Figure 2: Processing the results of the SLR of Step 1

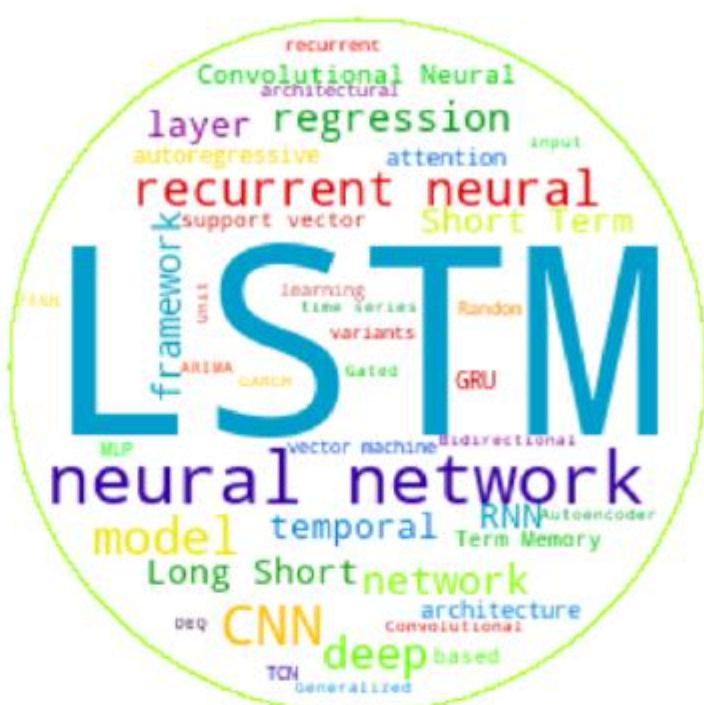

Figure 3: Word cloud illustrating frequently used sequential architectures

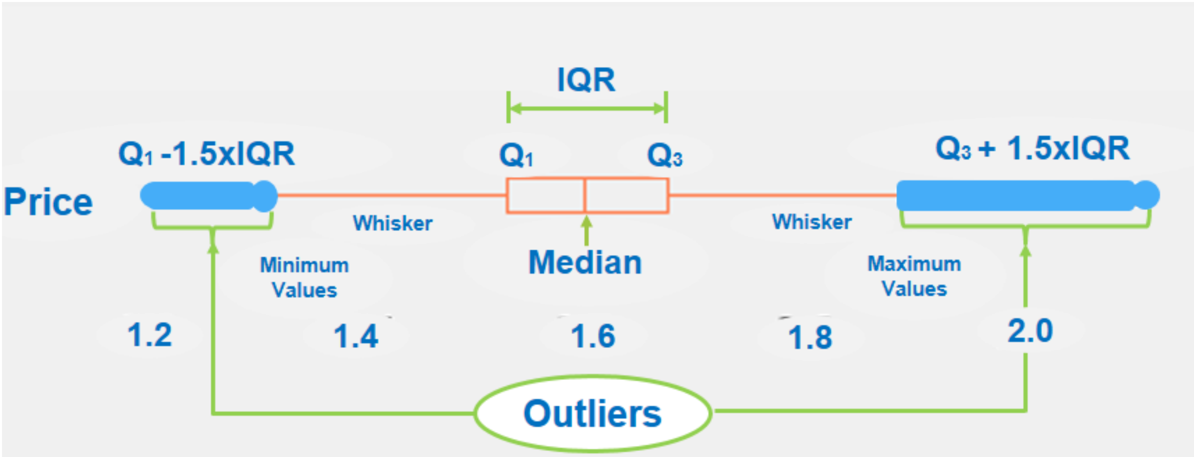

Figure 4: Box plot showing outlier distribution of daily exchange data between the GB Pound and the US Dollar from 1990 to 2016 of Step 2

Figure 5: IPPD visualization analysis of daily exchange data between the GB Pound and the US Dollar from 1990 to 2016 of Step 2

Lists 16 irregular sequences, found in a systematic literature review.

	List Name	Length	Number of IQR Outliers
1	S&P500 from Jan–Dec-2011 [136] and [21]	251	0
2	NASDAQ from Jan–Dec-2011 [136] and [21]	251	6
3	DJI from 01-2008–12-2009 [21]	504	0
4	NASDAQ from 01-2008–12-2009 [21]	504	0
5	S&P from 01-2008–12-2009 [21]	504	0
6	Monero Crypto Currency Daily Rates from 2015–2018 [39]	1208	0
7	DJI from 10-2010–09-2016 [9] and [21]	1513	0
8	S&P500 from 10-2010–09-2016 [9] and [21]	1696	0
9	CAD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
10	CNY-USD Daily Exchange Rate from 1990–2016 [22]	5000	0
11	NZD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
12	SGD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
13	JPY_USD Daily Exchange Rate from 1990–2016 [22]	5000	24
14	AUD_USD Daily Exchange Rate from 1990–2016 [22]	5906	0
15	GBP_USD Daily Exchange Rate from 1990–2016 [22]	6135	639
16	SwiFranc_USD DailyExchange Rate from 1990–2016 [22]	7015	0

Candidate deep learning models identified in Step 2 of the SeLFISA framework

Model	Architecture	Remarks
1	LSTM(32) + Dropout (0.2) + Dense (1)	Derived from Azlan et al. (2019) [5] and Li et al. (2019) [81]
2	LSTM(32) + LSTM(64) + Dropout(0.2) + LSTM(128) + Dropout(0.5) + Dense(1)	Influenced by Glenski et al. (2019) [39] and Chalvatzisa et al. (2019) [21]
3	Bi(LSTM(50)) + Dense(10) + Dense(10) + Dense(1)	A gated LSTM suggested by Sardelicha and Manandhara (2018) [102]
4	Bi(GRU(50)) + Dense(10) + Dense(10) + Dense(1)	A gated GRU mentioned by Sardelicha and Manandhara (2018) [102]
5	LSTM(100) + Dropout(100) + Attention(SeqSelfAttention(32)) + LSTM(16) + Dense(10) + Dense(10) + Dense(1)	Derived from experiments by Huang (2019) [54]
6	LSTM(32) + Conv1D(32) + Dropout(0.2) + Conv1D(16) + Conv1DTr(16) + Dropout(16) + Conv1D(32) + Conv1D (16) + Attention(SeqSelfAttention(1)) + LSTM(16) + Dropout(0.2) + Dense (1)	As indicated by Makinen et al. (2018) [77] and Huang (2019) [54]
7	LSTM(32) + Dropout(100) + Attention(SeqSelfAttention(32)) + LSTM(16) + Dense(10) + Dense(10) + Dense(1)	As implemented by Liu (2018) [74]
8	LSTM(32) + Dropout(0.2) + Attention(SeqSelfAttention(32)) + Bi(LSTM(32)) + Bi(LSTM(32)) + Dense(10) + Dense(1)	Demonstrated by Sardelicha and Manandhara (2018) [102]
9	LSTM(32) + Conv1D(32) + Dropout(0.2) + Conv1D(16) + Conv1DTranspose(16) + Dropout(0.2) + Conv1DTranspose(32) + Conv1DTranspose(1) + GRU(32) + Dropout(0.5) + Dense(1)	Suggested by Maggiolo and Spanakis (2019) [76]
10	GRU(32) + GRU(64) + Dropout(0.2) + GRU(128) + Dense(1)	Designed by GRU by Qin(2019) [97]
11	LSTM(32) + LSTM(64) + RepeatVector(64) + LSTM(64) + TimeDist(1) + LSTM(128) + Dropout(128) + Dense(1)	Suggested by Qin (2019) [97]
12	LSTM(50) + Dropout + LSTM(100) + Dropout(0.5) + GRU(100) + LSTM(100) + Dropout(0.5) + LSTM(100) + Dropout (0.5) + Dense(100) + Dense(10) + Dense(10) + Dense(1)	Implemented by Bai (2019) [7]

Results Of Deep Learning Framework Performance Evaluation Metrics Of Step 2

1. Agreement Cohen's Kappa
2. Average negative log-likelihood (NLL)
3. Computational time spent by a model
4. Copy memory loss and memory footprints
5. Correlation coefficient (R2)
6. Cosine proximity
7. Dynamic time warping (DTW)
8. Empirical correlation coefficient (CORR)
9. F-Measure
10. Hit ratio
11. Matthews correlation coefficient (MCC)
12. Max absolute percentage error (MaxAPE)
13. Mean absolute error (MAE)
14. Mean absolute percent errors (MAPE)
15. Mean absolute scaled error (MASE)
16. Mean directional accuracy (MDA)
17. Maximum error (ME)
18. Mean Error Percent (MEP)
19. Mean prediction accuracy (MPA)
20. Mean relative error (MRE)
21. Mean square error (MSE)
22. Mean squared percentage error (MSPE)
23. Mean symmetric mean absolute percentage error (SMAPE)
24. Median MASE
25. Median SMAPE
26. Normalized deviation (ND)
27. Normalized RMSE (NRMSE)
28. Normalized root mean squared error (NRMSE)
29. Precision F1 score
30. Precision jumps recall
31. Proportion of variance R2
32. Rank MASE
33. Rank SMAPE
34. Regression coefficient (R2)
35. Root mean square error (RMSE)
36. Root mean squared logarithmic error (RMLSE)
37. Root mean squared percentage error (RMSPE)
38. Root relative squared error (RRSE)
39. Symmetric mean absolute percentage error (SMAPE)
40. Trading profitability measures (cumulative return (CR), annualized return (AR), annualized volatility (AV), sharpe ratio and (SR) and draw-down (DD))

Figure 1: Framework evaluation metrics

Step 3: Implementation Algorithm

Algorithm 1: Implementation of models from core research articles and selection

```
Inputs--primary dataset split into training, and testing subsets; Validation dataset

For each model resulting from Step 2 of the SeLFISA framework:
    Get summary of model (design structure; total number of parameters)
    Set monitoring and regularization function
         to avoid overfitting (monitoring accuracy metric values)
    Fit model to training set using parameters recommended in articles
    Plot training performance loss to monitor training / validation loss
    Use trained model to generate predictions on the training set
    Evaluate training performance accuracy metrics

    Generate testing results on the testing subset:
        Use trained model to generate predictions on the testing set
        Record prediction accuracy metric results
        Visualise / Plot prediction results along with ground truth

    Generate testing results on the validation dataset:
        Use trained model to generate predictions on the validation set
        Record prediction accuracy metric results
        Visualise / Plot prediction results along with ground truth

    Save the model as .tf file
    Save output results as .csv file

Tabulate all results observations
Analyse the results
Select top N performing models as baseline models based on accuracy
         metric values on the testing portion of the primary dataset.
Compute efficiency and consistency of baseline models
```

Steps 5 and 6—Propose, design and implement a new artefact

Algorithm 2: Algorithm to arrive at an enhanced deep learning model

Inputs: Baseline models resulting from Step 3;
    primary dataset split into training set and testing set;
    validation dataset.
Part 1—Construct initial models based on the baseline models
Identify key combinations K of baseline models in Step 3 of the SeLFISA
    framework to enhance performance
For each combination M in K:
  Use the design of the models M as a basis for a new design EM
    placed in set $E = \{EM : M \in K\}$ of enhanced models
  Train the model EM
  Use EM to generate predictions on the testing set
  Record accuracy prediction metric results for EM
  Visualise / Plot prediction results along with ground truth

Part 2—Construct derivative models using
    baseline models and new models from Part 1
While performance of any model in E is not significantly better than baseline
    models and iterations $< N$:
  Execute Lines 5–11 of Algorithm 1, with M set to baseline models and
    models in E to create a new derivative model EM
  Append EM to E

  Select the highest–performing EM in E as the final enhanced model
  Apply optimisation to EM on additional hyperparameters, i.e. layers,
    activation functions, ordering, etc.
  Apply grid search optimisation to EM on additional hyperparameters,
    i.e. learning rate, dropout rate and batch_size.

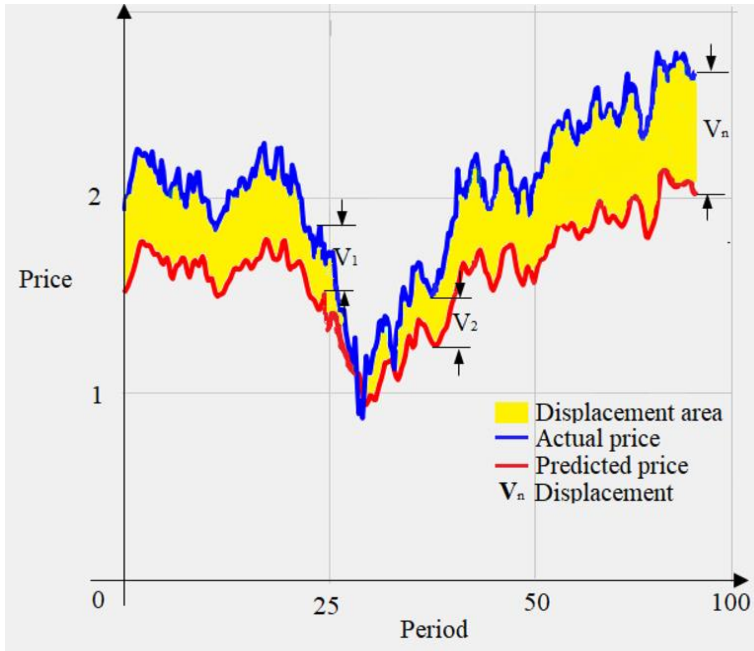

Figure 6: An illustration of the vertical displacement Vn from the ground truth prediction analysis of irregular patterns

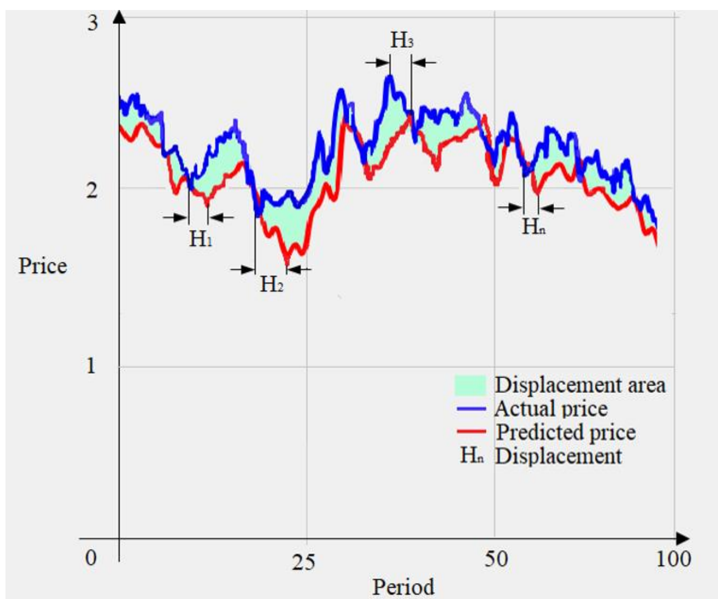

Figure 7: An illustration of the horizontal shift Hn from the ground truth prediction analysis of irregular patterns

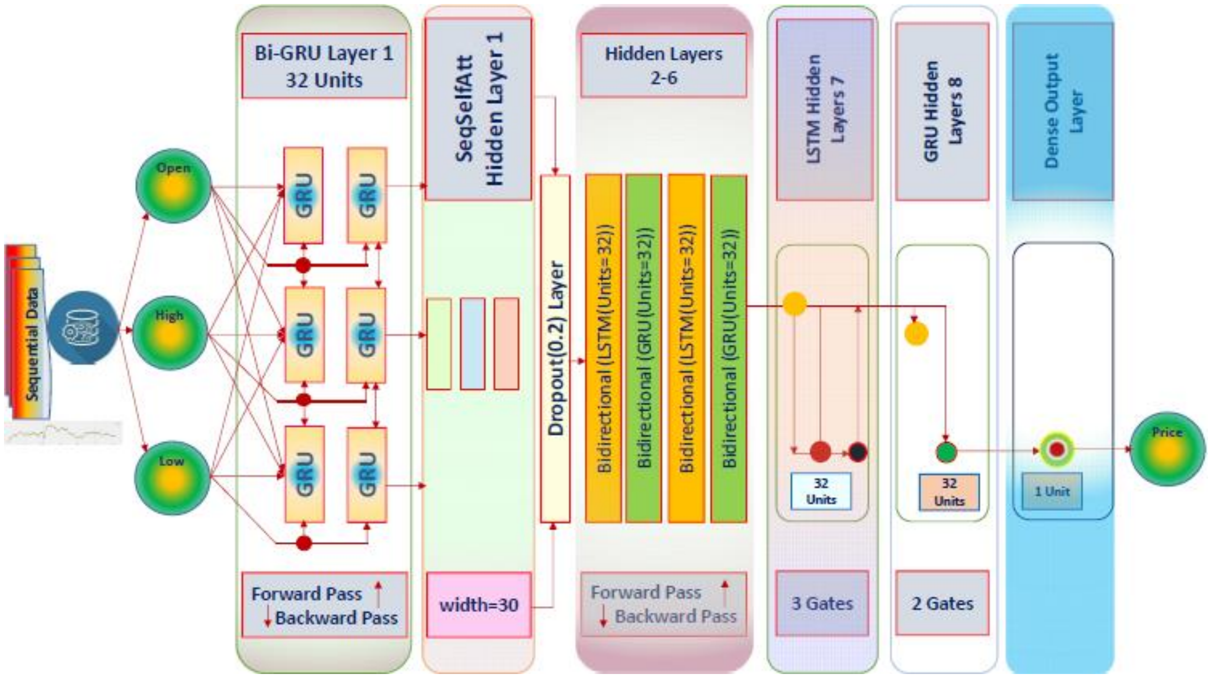

Figure 8: Enhanced Deep learning model of Steps 5 and 6 referred as the **Systematic enhanced deep Learning Framework for Irregular Sequential Analysis (SeLFISA) model**

Results of top-performing baseline models

Model	GBP/USD dataset			JPY/USD dataset			Training Efficiency			Consistency
	MAE	MSE	Adj. R^2	MAE	MSE	Adj. R^2	Number of Parameters	Time Seconds	Efficiency	
2	0.0487	0.00349	0.865	0.502	0.263	-5.15	128513	6430	**19.99**	0.61
3	**0.0167**	0.00311	**0.976**	0.172	0.0331	0.226	23131	1210	19.12	0.61
4	0.0321	0.00236	0.828	**0.0554**	**0.00561**	**0.362**	17931	963	18.62	**1.88**
7	0.0197	**0.00208**	0.885	0.345	0.127	-1.96	10276	3150	3.26	0.56

Results of the enhanced SeLFISA model

Model	GBP/USD dataset			JPY/USD dataset			Training Efficiency			Consistency
	MAE	MSE	Adj. R^2	MAE	MSE	Adj. R^2	Number of Parameters	Time Seconds	Efficiency	
SeLFISA	**0.0103**	**0.000255**	**0.981**	**0.0149**	**0.00333**	**0.421**	117538	4080	**28.81**	**2.74**
$\mathbf{P_i}$	47.41%	156.32%	0.51%	115.22%	51.01%	15.07%			36.15%	37.42%

Selected Papers that Analyse Sequences with Irregular Patterns

Lists papers that analyze irregular sequences, found in a systematic literature review.

List	SOTA Sequential Artefacts	Sequential datasets	Evaluation metrics
1.	Daily exchange rates data from Australia, British Canada, Switzerland, China, Japan, New Zealand and Singapore from 1990 to 2016 by (Lai et al. 2017)	Attention based frameworks (At-LSTM)	Agreement Cohen's Kappa
2.	NASDAQ stock price dataset by Qin Y. et al (2017)	Hybrid attention based frameworks (At-LSTM)	Average negative log-likelihood (NLL)
3.	Appliances energy prediction dataset by Candanedo L. et al (2017)	Autoregressive models (AR)	Computational time spent by a model
4.	Air quality prediction (AIR De Vito S. et al (2008)	Hybrid autoregressive model	Copy memory loss and memory footprints
5.	Weather dataset by Liang X, et al (2015)	Back-propagation neural networks (BPNN)	Correlation coefficient (R^2)
6.	European G´EANT traffic data points	Bayesian based algorithms	Cosine proximity
7.	Telecom datasets from Cell2Cell	Bidirectional (Bi) based frameworks	Dynamic time warping (DTW)
8.	Crowd Analytix dataset	Bidirectional combined with attention (Att) mechanism	Empirical correlation coefficient (CORR)
9.	Unstable social media dataset from Persian movie reviews from 2014 to 2016.	Bidirectional combined with GRU (BiGRU) and LSTM (BiLSTM)	F-Measure
10.	Standard benchmark ACL18 data for NASDAQ and NYSE markets from Jan 2014 to Jan 2016 by (Xu and Cohen, 2018)	Capsule neural network (CapsNet)	Hit ratio
11.	Standard KDD17 dataset by (Zhang et al., 2017)	Convolutional neural networks (CNNs)	Matthews correlation coefficient (MCC)
12.	Stock index data (DOW 30, S&P 500 and NASDAQ)	Deep autoencoder (DA)	Max absolute percentage error (MaxAPE)
13.	Ultra-high-frequency order book data from 5 liquid U.S NASDAQ's (Google, Microsoft, Apple, Intel and Facebook) financial stocks	Deep Bayesian neural networks (BNN)	Mean absolute error (MAE)
14.	Financial stock indices dataset (S&P 500, Dow Jones Industrial Average (DJIA), NASDAQ and Russel 2000)	Deep differential privacy-inspired LSTM (DP- LSTM)	Mean absolute percent errors (MAPE)
15.	Historical financial price data from Crypto-Compare for Bitcoin, Ethereum and Monero	Deep feed forward neural network (FFNN)	Mean absolute scaled error (MASE)
16.	Social data from publicly available social platforms (GitHub and Reddit).	Deep sequential spatio-temporal residual neural network (ST-ResNet)	Mean directional accuracy (MDA)
17.	Standard Penn Treebank (PTB) data	Denoising autoencoder (DAE)	Maximum error (ME)
18.	Standard WikiText-103 (WT103) data	Transformer neural network	Mean Error Percent (MEP)
19.	Financial news dataset from Reuters and Bloomberg on 473 Standard & Poor's 500 listed companies (Google, Amazon, Cisco, Microsoft, Apple, Intel, IMB, AMD, NVidia, Qualcomm, Walmart)	Transformer neural network combined with RNN and CNN	Mean prediction accuracy (MPA)
20.	Sydney motorway traffic flow data of 2017	TrellisNet	Mean relative error (MRE)
21.	Financial stock dataset from	Bank of China	(601988), Vanke

A (000002) and Kweichou Moutai (600519).	Differentiable architecture (DARTS)	Mean square error (MSE)

Behaviour of Irregular Sequential Patterns (Cont.)

List	SOTA Sequential Artefacts	Sequential datasets	Evaluation metrics
22.	UCI daily grocery sales datasets	Dilated recurrent neural network (DilatedRNN)	Mean squared percentage error (MSPE)
23.	Univariate (Daily values for Melbourne's minimum temperature and Zurich Sunspot) datasets	Dilated temporal convolutional network (TCN)	Mean symmetric mean absolute percentage error (SMAPE)
24.	Multi-variate (Energy production for 10 different photovoltaic power plants in California and SML2010 dataset containing internal and external measurements in a domestic house) datasets	Dual self-attention network (DSANet)	Median MASE
25.	Real time Yangtze River dissolved oxygen time series data automatically recorded from 2012 to 2016.	Dual-stage attention based recurrent neural network (DA-RNN)	Median SMAPE
26.	4 years sequential time series Uber dataset for 8 large cities in U.S. and Canada (Atlanta, Boston,Chicago, Los Angeles, New York City, San Francisco, Toronto, and Washington D.C)	Elmann recurrent neural networks (ERNN)	Normalized deviation (ND)
27.	Trajectory data (TaxiBJ from taxicab GPS data and meteorology data in Beijing (2013 – 2016) and Trajectory data (BikeNYC) from NYC bike system (2014)	Extension GARCH (EGARCH)	Normalized RMSE (NRMSE)
28.	Historical S&P 500 stock price data from the Yahoo Finance	Fast-slow recurrent neural network (FS-RNN)	Normalized root mean squared error (NRMSE)
29.	46. NLP sentimental news dataset from financial domain (CNBC.com, Reuters.com, WSJ.com, Fortune.com and Wall Street Journal)	Feed forward neural networks (FFNN)	Precision F1 score
30.	Daily revenue data from five gas stations companies	Generative adversary neural networks (GAN)	Precision jumps recall
31.	45 datasets of different time series lengths from random real world application domains which encompass Meteorology, Astronomy, Physiology, Acoustics, and others	Gated recurrent unit (GRU)	Proportion of variance R^2
32.	Real-world JD.com of China's (JD-demand and JD-shipment) data	Gated recurrent unit with hybrid architecture	Rank MASE
33.	Electricity consumption dataset for servers in a data centre by Flunkert et al.(2017)	Gaussian models (GP)	Rank SMAPE
34.	Traffic flows data by Lv et al. (2015)	General regression neural network (GRNN)	Regression coefficient (R^2)
35.	Internet traffic dataset for internet companies' by Kaggle (2017))	Generalized autoregressive conditional heteroscedasticity (GARCH)	Root mean square error (RMSE)
36.	Daily exchange rates data from Australia, British Canada, Switzerland, China, Japan, New Zealand and Singapore from 1990 to 2016 by (Lai et al. 2017)	Generalized linear regression (GLM))	Root mean squared logarithmic error (RMLSE)
37.	NASDAQ stock price dataset by Qin Y. et al (2017)	Hierarchical multi-scale recurrent neural network (HM-RNN)	Root mean squared percentage error (RMSPE)
38.	Appliances energy prediction dataset by Candanedo L. et al (2017)	Hierarchical neural network architecture	Root relative squared error (RRSE)
39.	Air quality prediction (AIR De Vito S. et al (2008)	Independently recurrent neural network (IndRNN)	Symmetric mean absolute percentage error (SMAPE)

Behaviour of Irregular Sequential Patterns (Cont.)

List	SOTA Sequential Artefacts	Sequential datasets	Evaluation metrics
40.	Weather dataset by Liang X, et al (2015)	Large feedforward neural network (LFNN)	Trading profitability measures (cumulative return (CR), annualized return (AR), annualized volatility (AV), sharpe ratio and (SR) and draw-down (DD))
41.	European G´EANT traffic data points	Logistic regression (LR)	
42.	Telecom datasets from Cell2Cell	Long short-term memory (LSTM)	
43.	Crowd Analytix dataset	Memory-based ordinal regression deep neural networks (MOrdReD)	
44.	Unstable social media dataset from Persian movie reviews from 2014 to 2016.	Momentum models (MOM)	
45.	Standard benchmark ACL18 data for NASDAQ and NYSE markets from Jan 2014 to Jan 2016 by (Xu and Cohen, 2018)	Mean reversion models (MR)	
46.	Standard KDD17 dataset by (Zhang et al., 2017)	Multilayer perception (MLP)	
47.	Stock index data (DOW 30, S&P 500 and NASDAQ)	Multivariate adaptive regression splines (MARS)	
48.	Ultra-high-frequency order book data from 5 liquid U.S NASDAQ's (Google, Microsoft, Apple, Intel and Facebook) financial stocks	Neural architecture search (NAS)	
49.	Financial stock indices dataset (S&P 500, Dow Jones Industrial Average (DJIA), NASDAQ and Russel 2000)	Particle filter recurrent neural networks (PF-RNNs)	
50.	Historical financial price data from Crypto-Compare for Bitcoin, Ethereum and Monero	Quasi-recurrent neural network (QRNN)	
51.	Social data from publicly available social platforms (GitHub and Reddit).	Radial basis neural networks (RBFNN)	
52.	Standard Penn Treebank (PTB) data	Random Classifier (RC)	
53.	Standard WikiText-103 (WT103) data	Random connectivity LSTM (RCLSTM)	
54.	Financial news dataset from Reuters and Bloomberg on 473 Standard & Poor's 500 listed companies (Google, Amazon, Cisco, Microsoft, Apple, Intel, IMB, AMD, NVidia, Qualcomm, Walmart)	Random forest (RF)	
55.	Sydney motorway traffic flow data of 2017	Recurrent highway network (RHN)	
56.	Financial stock dataset from Bank of China (601988), Vanke A (000002) and Kweichou Moutai (600519).	Recurrent neural network (RNN)	
57.	UCI daily grocery sales datasets	Rule-based regression (RBR)	
58.	Univariate (Daily values for Melbourne's minimum temperature and Zurich Sunspot) datasets	Sequence to sequence (Seq2seq) architectures or encoder-decoder models	
59.	Multi-variate (Energy production for 10 different photovoltaic power plants in California and SML2010 dataset containing internal and external measurements in a domestic house) datasets	Skip recurrent neural network (SkipRNN)	

Behaviour of Irregular Sequential Patterns (Cont.)

List	SOTA Sequential Artefacts	Sequential datasets	Evaluation metrics
60.	Real time Yangtze River dissolved oxygen time series data automatically recorded from 2012 to 2016.	Small feedforward neural network (SFNN)	
61.	4 years sequential time series Uber dataset for 8 large cities in U.S. and Canada (Atlanta, Boston,Chicago, Los Angeles, New York City, San Francisco, Toronto, and Washington D.C)	Spatio-temporal long short-term network (ST-LSTM)	
62.	Trajectory data (TaxiBJ from taxicab GPS data and meteorology data in Beijing (2013 – 2016) and Trajectory data (BikeNYC) from NYC bike system (2014)	Squares support vector machine regression (LS-SVMR).	
63.	Historical S&P 500 stock price data from the Yahoo Finance	StockNet which uses a variational autoencoder (VAE)	
64.	46. NLP sentimental news dataset from financial domain (CNBC.com, Reuters.com, WSJ.com, Fortune.com and Wall Street Journal)	Support vector machine regression (SVMR)	
65.	Daily revenue data from five gas stations companies	Support vector machines (SVM)	
66.	45 datasets of different time series lengths from random real world application domains which encompass Meteorology, Astronomy, Physiology, Acoustics, and others	Temporal convolutional networks (TCN)	
67.	Real-world JD.com of China's (JD-demand and JD-shipment) data	Transformer networks	
68.	CIF 2016 Forecasting Competition Dataset	TrellisNet	
69.	NN5 Forecasting Competition Dataset	Variational LSTM	
70.	M3 Forecasting Competition Dataset		
71.	M4 Forecasting Competition Dataset		
72.	CIF 2016 Forecasting Competition Dataset		
73.	NN5 Forecasting Competition Dataset		

Variables identified and used inthe SeLFISA Framework

Lists of 16 irregular sequences, found in a systematic literature review.

Table B.1: Behaviour of Irregular Sequential Patterns

	List Name	Length	Number of IQR Outliers
1	S&P500 from Jan–Dec-2011 [136] and [21]	251	0
2	NASDAQ from Jan–Dec-2011 [136] and [21]	251	6
3	DJI from 01-2008–12-2009 [21]	504	0
4	NASDAQ from 01-2008–12-2009 [21]	504	0
5	S&P from 01-2008–12-2009 [21]	504	0
6	Monero Crypto Currency Daily Rates from 2015–2018 [39]	1208	0
7	DJI from 10-2010–09-2016 [9] and [21]	1513	0
8	S&P500 from 10-2010–09-2016 [9] and [21]	1696	0
9	CAD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
10	CNY-USD Daily Exchange Rate from 1990–2016 [22]	5000	0
11	NZD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
12	SGD_USD Daily Exchange Rate from 1990–2016 [22]	5000	0
13	JPY_USD Daily Exchange Rate from 1990–2016 [22]	5000	24
14	AUD_USD Daily Exchange Rate from 1990–2016 [22]	5906	0
15	GBP_USD Daily Exchange Rate from 1990–2016 [22]	6135	639
16	SwiFranc_USD DailyExchange Rate from 1990–2016 [22]	7015	0

Variables identified and used inthe SeLFISA Framework

Variables for SeLFISA Framework

No.	Variable Name	Feature Count	Feature List	Category	Implementation Remarks
1.	Domain of research	1	Deep Learning Framework for the Prediction of Discrete Irregular Patterned Sequential Environments	Combinatorial in identification and selection but permutative in implementation	This is the initial stage driven by the research challenges in sequential modelling.
2.	Domain Challenges classes	3	Major classes are within frameworks, datasets and evaluation	Combinatorial	Addressed all through a framework
3.	Specific prediction challenges	11	consistency or inconsistency, reliability, repeatability, straightforwardness transparency, explainability, sensitivity to outliers and extreme values, lack of well-established, explainable literature, poor comprehensive comparison analysis, lack of multidimensional performance evaluation on single framework, dominance of accuracy metrics, computational complexity.	Combinatorial	Focused on those that distort performance robustness
4.	Existing research sources	400	400 articles were the initial sources of literature research	Combinatorial	33 articles created nucleus articles based on a matrix specific selection, inclusion and analysis criteria.
5.	Implementation AI Platforms	8	Google AI Cloud Platform, Amazon AI Services (Amazon SageMaker), Google Cloud AutoML, MATLAB, Microsoft Azure (Machine Learning Studio), IBM Watson Machine Learning and Anaconda Enterprise.	Combinatorial	Anaconda Enterprise was our platform of choice because its open source versatility platform with a Python based IDE compatibility with many languages and notebooks. This avail the entire life cycle which prepare, build, validate, deploy and monitor AI models.
6.	Implementation Languages	5	Python, Java, Lisp, Prolog and R Programming	Combinatorial	Python was our language of choice since it is easy to learn, deploy and it integrates efficiently with a wide range of syntax

Variables for SeLFISA Framework (Cont.)

No.	Variable Name	Feature Count	Feature List	Category	Implementation Remarks
7.	Implementation environ-ments	3	Jupyter Notebook, Kaggle and Google Colaboratory	Combinatorial	We created our environment based on Jupyter Notebook be-cause of its interactive features that can mix code, script, inline graphs, interactive figures, into a shareable web document.
8.	Libraries and modules	22	Regular expressions, garbage collectors, operating systems, system-specific parameters, time, spacy, Keras, pickle, requests, math time, Matplotlib, NumPy, Pandas, progress bar TQDM library, math log2, Seaborn, sklearn, metrics, TensorFlow.	Combinatorial	We choose more than 22 li-braries and modules that are already written in Python to set routines and functions. These libraries and modules were expanded from internal module through an "from main library import internal library"
9.	Computational Environment	4	High Performance Computing from CHPC, Google Cloud, Kaggle and On-Premise Core i7 Laptop.	Combinatorial on design and Permutative on installation and executions	CHPC High Performance Com-puting combined through on-Prem Laptop.
10.	Datasets domain	8	Weather, energy, finance, weather, astronomy, transportation, health and general domain benchmark datasets.	Combinatorial	Finance domain was our pri-mary choice.
11.	Datasets	73	The 8 domains from 33 nucleus articles produced 73 accessible datasets.	Combinatorial	2 Financial market-daily cur-rency exchange datasets were selected with high levels of irregular discrete properties.
12.	Selected dataset features	6	Date, price, open, high, low and change	Combinatorial	Pre-process before training.
13.	Data ex-ploratory processes	10	More than 10 activities in the form of data wrangling, descrip-tion, data pre-processing, data munching, data cleaning, and exploratory data analysis	Permutative	Pre-process before training.
14.	Dataset splitting ratio	3	Training, validation and testing (80%–20%, 90%–10% and 70%–30%) and respective window length to determine prediction horizon.	Combinatorial on ratio se-lection and permutative on execution	Pre-process before training. A window length of an array of 100 inputs were used to determine the next outcome.
15.	Algorithms and models	335	These architectures produced 335 algorithms and models based on statistical, probabilistic, gated, attention, bidirectional, general neural, encoders and decoders, transformer, vanilla, hybrid, ensemble, convolutional, classifi-cation and other	Combinatorial on selection and Permu-tative during execution	Focused on best performing through experimental deploy-ment and application
16.	Algorithms learning technique	4	Supervised, Semi supervised, Unsupervised and Reinforcement	Combinatorial but learning process is per-mutative	Pre-process before training.
17.	Algorithms analysis types	4	Regression, classification, cluster-ing and association	Combinatorial	Regression analysis was ap-plied
18.	Evaluation criteria category	2	Quantitative and Qualitative	Combinatorial	Pre-process before training

Variables for SeLFISA Framework (Cont.)

No.	Variable Name	Feature Count	Feature List	Category	Implementation Remarks
19.	Evaluation criteria	9	Consistency, efficiency, accuracy, visualization sharpness, computational complexity. repeatability, straightforwardness and explainability	Combinatorial	Pre-process before training
20.	Activation function	24	ReLU. Leaky ReLU, Maxout, Tanh, linear / identity, Binary step, piece wise linear, Sigmoid, Complementary log-log, Bipolar, Bipolar Sigmoid, LeCun's Tanh, Hard Tanh, Absolute, Rectifier, Smooth Rectifier, Logit, Probit, Cosine, Softmax, Maxout, Multiquadratic and Inverse Multiquadratic.	Pre-process before training	
21.	Evaluation metrics category	3	Regression, binary classification and multi-class classification	Combinatorial	Regression was the choice of the research
22.	Evaluation metrics or Loss functions	12	Mean Error (ME),Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE),R Squared, Categorical Cross Entropy, Binary Cross Entropy ,Hinge Loss, Squared Hinge, Multi-Class Cross-Entropy Sparse Multi-class Cross-Entropy and Kullback Leibler Divergence.	Combinatorial	A hybrid approach was considered
23.	Weights	1	Randomly allocated using parameter optimisation techniques	Permutative	Automatically assigned
24.	Bias	1	Automatically selected using libraries.	Permutative	Guided by other factors
25.	Net input	1	Depend on the nature of the input features of the dataset.	Permutative	Guided by other factors
26.	Number of Neurons	1	Determined by a specific mathematical formula	Permutative	Guided by other factors
27.	Number of layers	1	Determined by a specific mathematical formula	Permutative	Guided by other factors
28.	Interconnections	2	Feed-forward and recurrent Combinatorial and permutative	Guided by other factors	
29.	Training process	2	Backpropagation and Backpropagation through time		Automatically implemented through Python libraries.
	Number of subvariables	≥ 410			

Visualisation of Models Predicting JPY vs. USD

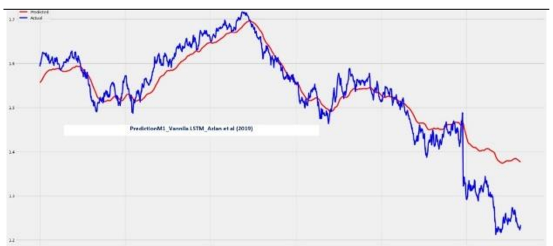

Figure D.1: Model 1

LSTM (32) + Dropout (0.2) + Dense (1) suggested by Azlan et al(2019) [5], Li et. al (2019) [81] and Glenski et. al (2019) [39]

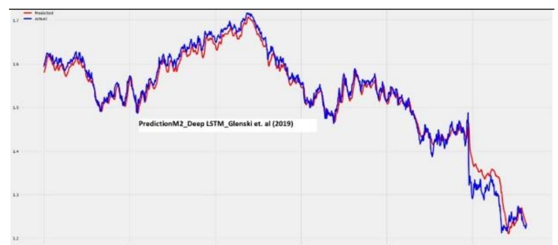

Figure D.2: Model 2

LSTM(32) + LSTM(64)+Dropout(0.2) + LSTM(128) + Dropout(0.5) + Dense(1) by Deep LSTM Model based implemented by Glenski et. al (2019) [39] and Chalvatzisa et. al (2019) [21]

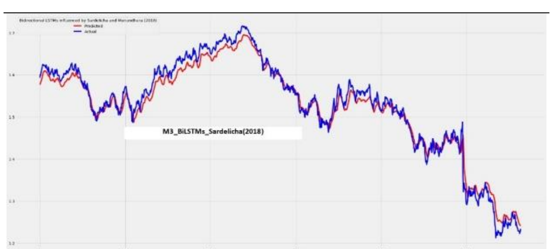

Figure D.3: Model 3

BiD(LSTM(50)) + Dense(10) + Dense(10) + Dense(1) influenced by Sardelicha and Manandhara (2018) [102]

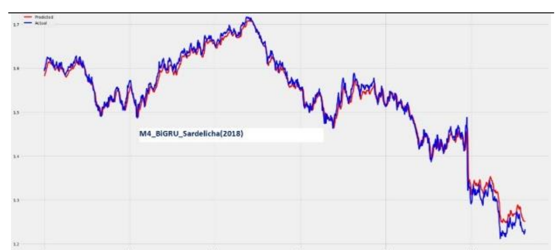

Figure D.4: Model 4

BiD(GRU(50)) + Dense(10) + Dense(10) + Dense(1) by Sardelicha and Manandhara (2018) [102]

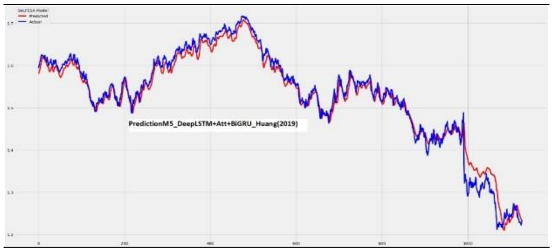

Figure D.5: Model 5

LSTM(100) + Dropout(100) + Attention(SeqSelfAttention) + LSTM(16) + Dense(10) + Dense(10) + Dense(1) by Deep LSTM by Huang (2019) [54]

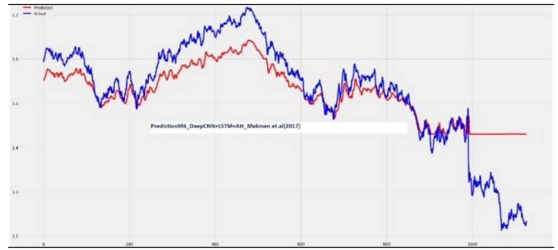

Figure D.6: Model 6

LSTM(32) + Conv1D(32) + Dropout(0.2) + Conv1D (16) + Conv1DTr(16) + Dropout(16) + Conv1DTr(32) + Conv1D (16) + AttSeqSelf(1) + LSTM(16) + Dropout(0.2) + Dense (1) by Makinen et. al (2017) [77]SeqSelf(1) + LSTM(16) + Dropout(0.2) + Dense (1) attention by Huang (2019) [54]

Figure D.7: Model 7

LSTM(32) + Dropout(100) + Attention (SeqSelf (32)) + LSTM (16) + Dense(10) + Dense(10) + Dense(1) by Liu (2018) [74]

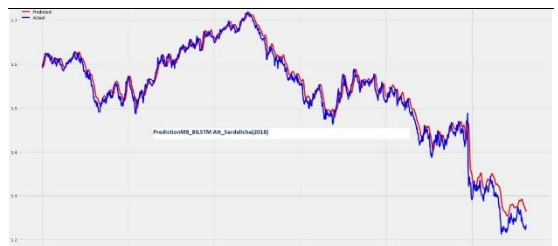

Figure D.8: Model 8

LSTM(32)+Dropout(0.2) + Attention (SeqSelf)(32) + Bidirection(LSTM(32)) + Bidirection(LSTM(32)) + Dense(10) + Dense(1) by by Sardelicha and Manandhara (2018) [102]

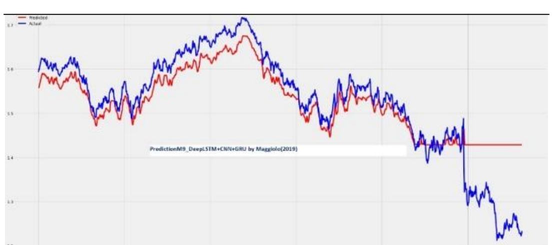

Figure D.9: Model 9

LSTM(32) + Conv1D(32) + Dropout(0.2) + Conv1D(16) + Conv1DTranspose(16) + Dropout(0.2) + Conv1DTranspose(32) + Conv1DTranspose(1) + GRU(32) + Dropout(0.5) + Dense(1) by Maggiolo and Spanakis (2019) [76]

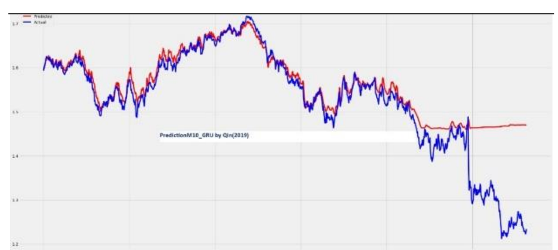

Figure D.10: Model 10

GRU(32) + GRU(64) + Dropout(0.2) + GRU(128) + Dense(1) by GRU by Qin(2019) [97]

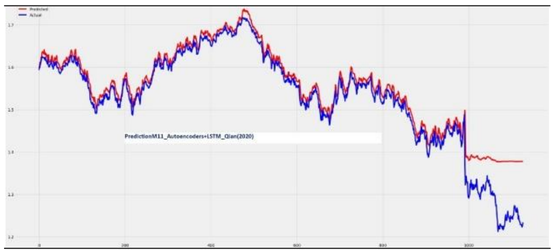

Figure D.11: Model 11

LSTM(32) + LSTM(64) + RepeatVector(64) + LSTM(64) + TimeDist(1) + LSTM(128) + Dropout(128) + Dense(1) by Qin(2019) [97]

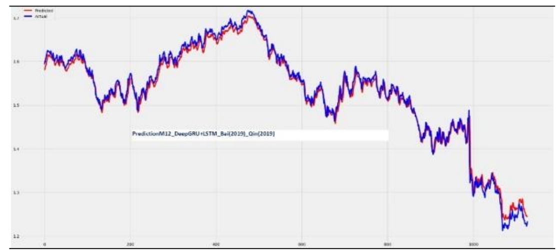

Figure D.12: Model 12

LSTM(50) + Dropout + LSTM(100) + Dropout (0.5)+ GRU(100) + LSTM(100) + Dropout(0.5) + LSTM(100) + Dropout(0.5) + Dense(100) + Dense(10) + Dense(10) + Dense(1) by Bai(2019) [7]

Figure D.13: SeLFISA Model or Enhanced Model

BiD(GRU(32)) + SeqSelfAtt(att width=30) + Dropout(0.2) + BiD(LSTM(32)) + BiD(GRU(32)) + BiD(LSTM(32)) + BiD (GRU(32)) + LSTM (32) + GRU(32) + Dense(1)