Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi

Afyon Kocatepe University Journal of Science and Engineering

AKÜ FEMÜBİD **23** (2023) 035104 (677-692)   AKU J. Sci. Eng. **23** (2023) 035104 (677-692)

DOI: 10.35414/akufemubid.1175297

Araştırma Makalesi / Research Article

Gated Recurrent Unit Network-based Fuzzy Time Series Forecasting Model

Serdar ARSLAN[1]

[1]*Cankaya University, Faculty of Engineering, Computer Engineering Department, Ankara.*

e-mail: sarslan@cankaya.edu.tr.   *ORCID ID: https://orcid.org/0000-0003-3115-0741*

Geliş Tarihi: 14.09.2022   Kabul Tarihi: 02.05.2023

Keywords
Gated Recurrent Unit; Time Series Forecasting; Fuzzy Time Series; Deep Learning.

Abstract

Time series forecasting and prediction are utilized in various industries, such as e-commerce, stock markets, wind power, and energy demand forecasting. An accurate forecast in these applications is an essential and challenging task because of the complexity and uncertainty of time series. Nowadays, deep learning methods are popular in time series forecasting and show better performance than classical methods. However, in the literature, only some studies use deep learning methods in fuzzy time series (FTS) forecasting. In this study, we propose a novel FTS forecasting model based upon the hybridization of Recurrent Neural Networks with FTS to deal with the complexity and uncertainty of these series. The proposed model utilizes Gated Recurrent Unit (GRU) to make predictions using a combination of membership values and past values from original time series data as model input and produce real forecast value. Moreover, the proposed model can handle first-order fuzzy relations and high-order ones. In experiments, we have compared our model results with state-of-art methods by using two real-world datasets; The Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) and Nikkei Stock Average. The results indicate that our model outperforms or performs similarly to other methods. The proposed model is validated using the Covid-19 active case dataset and BIST100 Index dataset and performs better than Long Short-term Memory (LSTM) networks.

Kapılı Tekrarlayan Hücreler Tabanlı Bulanık Zaman Serileri Tahminleme Modeli

Anahtar kelimeler
Kapılı Tekrarlayan Hücreler; Zaman Serisi Tahminleme;Bulanık Zaman Serisi; Derin Öğrenme.

Öz

Zaman serisi tahminleme hava durumu, iş dünyası, satış verileri ve enerji tüketimi tahminleme gibi bir çok alanda uygulama alanına sahiptir. Bu alanlarda tahminleme yaparken kesin sonuçlar elde etmek çok önemlidir ama aynı zamanda zaman serilerinin karmaşık ve de belirsizlik içeren veriler olması nedeniyle çok zordur. Günümüzde, derin öğrenme metotları bu alanda klasik metotlara göre daha iyi sonuçlar vermektedir. Fakat literatürde bulanık zaman serileri tahminleme konusunda çok az çalışma vardır. Bu çalışmada, zaman serilerindeki karmaşıklığın ve belirsizliğin doğurduğu problemleri yok etmek için Yinelemeli sinir Ağları ile bulanık zaman serilerini bir arada kullanan bir model ortaya konumuştur. Bu çalışmada, Kapılı Tekrarlayan Hücreler kullanarak geçmiş veriler ile bulanık verilerin üyelik değerleri birleştirilerek tahminleme değeri hesaplanmıştır. Ayrıca, bu çalışmadaki model ilk seviye bulanık ilişkileri ele alabildiği gibi, çoklu seviye bulanık ilişkileri de kapsamaktadır. Testlerde literatürde var olan çalışmalar ilgili model ile iki açık veri seti ile karşılaştırılmış olup bahsi geçen modelin daha iyi veya benzer sonuçlar verdiği gözlemlenmiştir. Ayrıca model Covid-19 ve BIST100 borsa verileri kullanılarak da test edilmiş ve Uzun-Kısa Süreli Bellek modellerinden daha iyi sonuç vermiştir.

1. Introduction

Time series analysis and forecasting methods are important research areas in machine learning and indispensable in diverse applications such as energy consumption, business retail or the stock market, cryptos, wind power, antibiotic resistance, weather pollution, water pollution, etc. In a simple manner, time series forecasting is a process of predicting the following values of time series by examining its historical values. Time series forecasting aims to understand and extract some implicit patterns such as seasonality, trend, and noise in these series. However, the traditional time series approaches cannot solve prediction problems on data with uncertainty raised due to imprecision, vagueness, and representation of this series in linguistic terms. Thus, Fuzzy Time Series (FTS) was proposed by Song and Chissom in 1993 (Song and Chissom 1993) in order to cope with this uncertainty problem by using the concepts of fuzzy set theory. In FTS, historical data is represented as linguistic values. The domain of the time series dependent variable, Universe of Discourse (*U*), is first defined and then divided into sub-domains. These sub-domains are linked to a fuzzy set, and this process is called *the fuzzification* step. After *the fuzzification* step, the fuzzy logical relationships (FLR) are established in *the fuzzy inference* step. In this step, forecasts are produced using FLRs. As a final step, after FLR identification and obtaining the forecast values on FTS, *the defuzzification method is applied* to have actual forecast values.

The establishment of fuzzy relationships is very important in FTS since it directly affects the accuracy of the forecasting model (Yu and Huarng 2010, Panigrahi and Behera 2020). Thus, in the literature, there are many approaches proposed such as fuzzy logic group relation table (Song and Chissom 1993, Bulut 2014, Chen and Chen 2014, Chen and Tanuwijaya 2011, Efendi, Ismail, and Deris 2015, Huarng 2001b, Huarng 2001a, Lee, Wang, and Chen 2008), fuzzy relation matrices (Aladag *et al*. 2012; Sullivan and Woodall 1994; Tsaur, O Yang, and Wang 2005; Wong, Tu, and Wang 2010), statistical methods (Cai *et al*. 2015, Chang *et al*. 2011, George E. P. Box 2015, De Gooijer and Hyndman 2006, Kocak 2017, Novák 1995, Sadaei *et al*. 2016, Sadaei *et al*. 2019, Torbat *et al.* 2018, Tseng *et al*. 2001) in order to determine and establish these relationships.. In addition to these methods, there are also many machine learning methods for identifying FLRs. In (Bas *et al*. 2018), a high-order neural network based on pi-sigma artificial neural network is proposed for fuzzy time series. A single multiplicative neuron model is proposed in (Aladag 2013). In (Bas *et al*. 2015), a FTS neural network model is presented for both linear and non-linear time series. In (Cagdas *et al*. 2009, Aladag *et al.* 2010a, Egrioglu *et al*. 2009, Egrioglu *et al.* 2013, Huarng and Yu 2006), feed forward neural networks are used. These models used fuzzy sets index number as input and tried to produce an index number as output. However, (Yu and Huarng 2010) and (Aladag 2013) have used neural network model with different input/output. They used membership degrees to determine fuzzy relations. In (Hájek and Olej 2017, Kocak et.al. 2021), intuitionistic fuzzy time series are modeled with neural networks and membership degrees and non-membership degrees are used as input. Using membership degrees as input-output provides better accuracy results but requires much more computational time because of large number of nodes (Bose and Mali 2019). There are also other machine learning methods for determining FLRs such as (Cagcag *et al.* 2017, Chang *et al.* 2011, Chen and Kao 2013, Jang 1993, Nie 1997, Panigrahi and Behera 2020, Stefanakos 2016).

Recurrent Neural Network (RNN) is a type of deep learning method that uses sequential data or time series data (Lin *et al.* 1998). Long short-term memory networks(LSTM) and gate recurrent unit networks(GRU) are two popular variants of RNN with long-term memory. LSTM is an improved version of RNN by adding two extra gates to capture long-term information; thus, it performs better in long-sequence data. Bidirectional LSTM (BiLSTM) is a popular variant of LSTM that uses forward and backward LSTM layers to learn long-term bidirectional dependencies of time series. BiLSTM often performs better than LSTM (Siami-Namini, Tavakoli, and Namin 2019).

LSTM is widely used in time series forecasting (Lindemann *et al.* 2021, Arslan 2022). However, a few studies in the literature applied LSTM in FTS. In (Tran *et al.* 2018), multivariate fuzzy time series are used as input for their LSTM model for forecasting. In (Kocak *et al.* 2021), authors proposed an intuitionistic fuzzy time series (Castillo *et al.* 2007) forecasting model using a simple LSTM with a single hidden layer of LSTM units and an output layer used to make a prediction. They used merged membership and non-membership values as input to produce one-dimensional output.

Gated Recurrent Unit (GRU) is another RNN model that is very similar to LSTM with a few internal changes (Cho *et al.* 2014). It has considered having a simpler architecture than LSTMs since it uses only two gates, whereas LSTM has three gates. Similar to LSTMs, GRUs have also been used in forecasting time series in different types of domains (Becerra-Rico *et al.* 2020, Dutta *et al.* 2020, Shen *et al.* 2018a, Tan *et al.* 2020, Wang *et al.* 2018) and outperforms LSTM models (Jozefowicz, Zaremba, and Sutskever 2015, Yang *et al.* 2020). However, these studies use classical time series for forecasting.

Recently, hybrid models which use a neural network with FTS are a significant development in forecasting (Singh 2017). Using deep learning methods provides the ability to deal with large data sets and also ensures fast learning capability. On the other hand, enabling fuzzy sets in the forecasting domain contributes to handling uncertainty and imprecision of time series data. Thus, hybridizing deep learning models with FTS is expected to produce much better results than classical forecasting methods, especially for complex systems.

In this study, we used GRU and BiLSTM with FTS (called FTS-GRU and FTS-BiLSTM, respectively) to identify first-order and high-order fuzzy logical relationships and compared the proposed hybrid model with neural network models, including LSTM. Moreover, the hybrid model is compared with other state-of-art models in the literature. The input of the proposed model is fuzzy membership values to all fuzzy sets. The model is trained, the optimal weights for fuzzy relations are obtained, and the model output is computed using the trained model. The accuracy of the model is calculated using outputs and actual values. Hence the main contributions of this work are enlisted as follows;

- Two different RNN-based deep learning methods, GRU (FTS-GRU) and BiLSTM (FTS-BiLSTM) are separately used for defining FLRs that support first-order FTS and high-order FTS.
- The fuzzy time series model obtained using GRU for identifying fuzzy relations using membership values as input and output produces superior forecasting performance than other RNN-based approaches.
- An empirical study has been conducted for four real-world data sets; TAIEX, Nikkei 225 stock exchange, BIST100 index, and Covid-19 cases in Turkey.
- The proposed models are also compared with state-of-art methods, and the FTS-GRU hybrid model shows better forecasting performance than these methods.

2. Methods

2.1. LSTM

LSTM neural networks are designed as a new recurrent neural network (RNN) form. Essentially, LSTM networks have their memory structure. Typical RNNs endeavor to solve the poor performance of feed-forward neural networks on sequential inputs. They are widely used in speech recognition, opinion and sentiment analysis, text processing, and time series prediction. In the LSTM model, an extended data sequence is memorized or held by adapting a gating structure. The structure of an LSTM cell is depicted in Fig. 1.

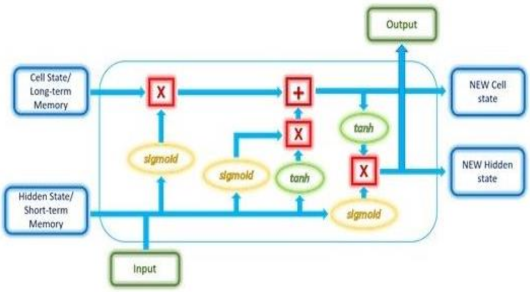

Fig. 1. LSTM cell structure (Arslan 2022)

The forget gate is utilized to specify which data will be preserved or not. In order to achieve this preservation, the following formula are used;

$$f_t = \sigma\,(W_f[\,h_{t-1}, x_t] + b_f) \quad (1)$$

where x_t is input at time t, h_{t-1} is the output of previous cell, and σ is sigmoid function. The information is kept in the cell state if forget gate produces one as output. In the next stage, the sigmoid function constructs a vector. This vector contains possible new values. Input gates are used to specify the updated values, and new possible values are stored in the vector $C^{\sim}{}_t$. This new vector is constructed with the following formulas;

$$i_t = \sigma(W_i[\,h_{t-1}, x_t] + b_i) \quad (2)$$

$$C^{\sim}{}_t = \tanh(\,W_c[\,h_{t-1}, x_t] + b_c \quad (3)$$

Now cell's old state C_{t-1} is updated to new cell state C_t.

$$C_t = f_t * C_{t-1} + \; i_t * C^{\sim}{}_t \quad (4)$$

Eventually, we select the network's output regarding on the cell state. This selection process is carried out by using the following formulas;

$$o_t = \sigma(W_o[\,h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

Stacked LSTMs (SLSTMs) are a special type of classical LSTMs (Graves *et al.* 2013). In SLSTMs, one and/or more LSTM layers are utilized and merged. The first LSTM layer employs the time series data as input and deliver the output. This output now becomes the input of next LSTM layer. All LSTM layers have an identical inner architecture with various units. Fig. 2 represents an example SLSTM network.

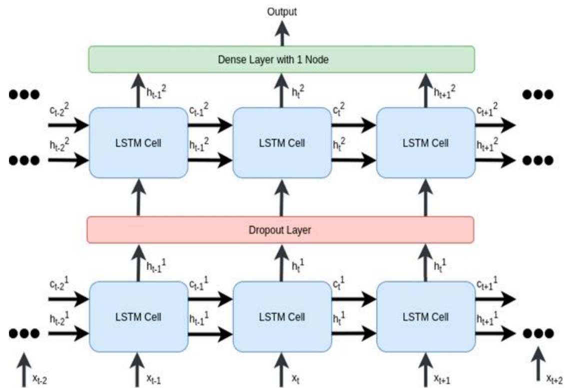

Fig. 2. Stacked LSTM network (Arslan 2022)

Classic LSTMs use only previous information to resolve the following states. Bidirectional LSTMs (BiLSTMs) are devised to deal with data in both directions (Schuster and Paliwal 1997). Two distinct hidden layers assemble BiLSTMs. Therefore, in order to push bidirectional information passing achievable at every time step, the overall model uses these two separated LSTMs. A BiLSTMs cell has two inputs; one from the previous step and the other from the next step. These two inputs are fused in the cell; thus, it makes BiLSTMs network capable of storing information from both the past and future. The general architecture of BiLSTM is illustrated in Fig. 3 (Cui and Wang 2017).

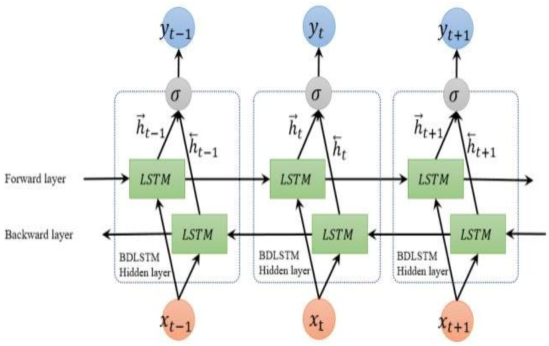

Fig. 3. Bidirectional LSTM network (Arslan 2022)

2.2. GRU

GRU is simplified version of LSTM and widely used in sequence modeling, natural language processing, time series analysis etc. Unlike LSTM, GRU has only two gates; update and reset. Update gate is used to determine the amount of previous data which will be passed to the next state. By using update gate, GRU model can copy all the past data if necessary. Reset gate is, on the contrary to update gate, for forgetting or neglecting the past data. This means, with the help of reset gate, GRU model decides whether previous cell state is important or not.

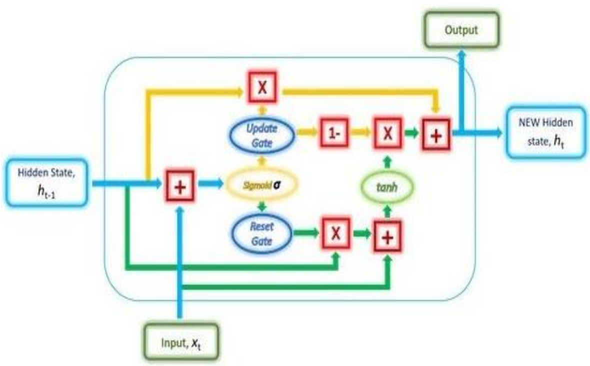

Fig. 4. GRU cell structure (Arslan 2022)

GRU models do not contain internal memory and also do not have output gate which is used in LSTM models (Fig. 4). The recent studies shows that the GRU outperforms the LSTM on forecasting tasks (Dutta *et al.* 2020; Jozefowicz *et al.* 2015; Shen *et al.* 2018b).

2.3. Fuzzy Time Series

Fuzzy Time Series(FTS) are first introduced in (Song and Chissom 1993) and uses basic fuzzy set principles developed by Zadeh (Zadeh 1965). For last decades, FTS have been applied to various number of forecasting problems in the literature (Bose and Mali 2019, Singh 2017). A fuzzy set can be described as class with varying degrees of membership in the set.

Let U be the universe of discourse, where

$$U = \{u_1, u_2, \dots, u_n\} \tag{7}$$

Thus, a fuzzy set *A* of the universe of discourse *U* is defined as follows:

$$A = f_A(u_1)/u_1 + f_A(u_2)/u_2 + \dots + f_A(u_n)/u_n \tag{8}$$

where, f_A is the membership function of the fuzzy set *A*, $f_A : U \rightarrow [0, 1]$ and $f_A(u_i)$ represents the degree to which u_i belongs to the fuzzy set A, and $1 \leq i \leq n$. In this equation "$+$" symbol represents union operation. If the universe of discourse *U* is infinite and also continuous then the fuzzy set *A* can be defined as:

$$A = \{\int f_A(u_i)/u_i\} \tag{9}$$

where $\forall u_i \in U$ and "∫" symbol represents union of fuzzy singletons $f_A(u_i)/u_i$.

The definitions of fuzzy time series are reviewed as follows:

Definition 2.1. Let $Y(t)(t = \cdots, 0,1,2, \dots)$ be a subset of R (real number) and also the universe of discourse on which fuzzy sets $f_i(t)$ $(i = 1,2, \dots)$ are defined. Let $F(t)$ be a collection of $f_i(t)$ $(i = 1,2, \dots)$. Then, $F(t)$ is called a fuzzy time series on $Y(t)(t = \cdots, 0,1,2, \ldots)$ (Song and Chissom 1993).

In this definition, $F(t)$ represents a linguistic variable and $f_i(t)$ $(i = 1,2, \ldots)$ can be regarded as possible linguistic values of $F(t)$ (Song and Chissom 1993).

Definition 2.2. Let U_{min} and U_{max} be minimum and maximum values of the time series data respectively. By using these values, we can define the universe of discourse *U* as (Song and Chissom 1993):

$$U = [\, U_{min}, U_{max}] \tag{10}$$

Thus, fuzzy logical relationships can be established using the following definition.

Definition 2.3. If there exist a fuzzy relationship $R(t-1,t)$, such that:

$$F(t) = F(t-1) \; o \; R(t-1,t) \tag{11}$$

Then $F(t)$ is said to be caused by $F(t-1)$. "o" symbol represents max-min composition operator. Assume that $F(t-1) = A_i$ and $F(t) = A_j$. The relationship between $F(t)$ and $F(t-1)$ is called as a fuzzy logical relationship (FLR) and can be represented by:

$$A_i \rightarrow A_j \tag{12}$$

In Eq.12, A_i refers to left-hand side (LHS) and A_j refers to right-hand side (RHS) of FLR. Now we can group FLRs if FLRs have the same fuzzy sets on LHS and this group is referred as fuzzy logical relationship group (FLRG). For example, if we have following FLRs;

$$A_i \rightarrow A_k$$
$$A_i \rightarrow A_l$$
$$A_i \rightarrow A_m$$

These FLRs can be grouped into a FLRG as:

$$A_i \rightarrow A_k, A_l, A_m \tag{13}$$

FTS models are represented by using these FLRs. If $F(t)$ is caused by only $F(t-1)$ then it is called first-order model and can be represented as Eq.11. However, if $F(t)$ is caused by $F(t-1)$, $F(t-2)$,..., and $F(t-n)$, then this model is called n-th order model and can be represented as:

$$F(t-n), \ldots, F(t-2), F(t-1) \rightarrow F(t) \tag{14}$$

2.4. FTS forecasting model

FTS forecasting models uses five different steps in general (Panigrahi and Behera 2020, Bose and Mali 2019, Singh 2017, Chen 1996); *defining and partitioning the universe of discourse*, *defining fuzzy sets*, *fuzzification of data*, *identifying of FLRs and FLRGs*, *forecasting and defuzzification* and these steps shown in Fig. 5.

Step I. Defining the universe of discourse. The universe of discourse *U* can be defined by using Eq.10. In order to partition *U*, the length of the intervals "*l*" should be defined. After that, U is partitioned into equal length intervals by using this *l* value. There are several studies in the literature for finding intervals on *U* and effective partitioning (Panigrahi and Behera 2020, Bose and Mali 2019, Singh 2017).

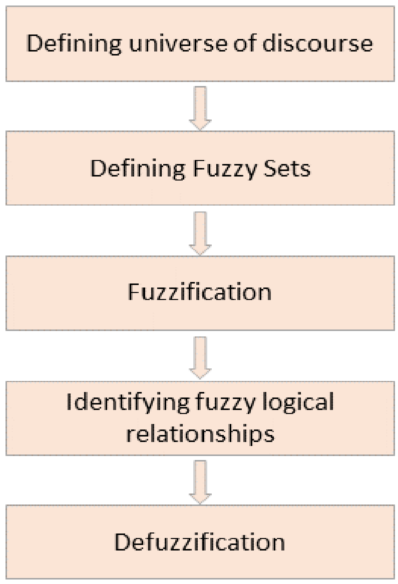

Fig. 5. FTS forecasting model steps

For an example, U can be defined as [4600,10300] for TAIEX 2000 dataset. After finding interval length *l,* U is divided into equal-lengths partitions. The number of partitions called "*n*" can be calculated by using following equation:

$$n = \frac{(U_{max} - U_{min})}{l} \quad (15)$$

Let *l* is equal to 100, then *U* has *n* = 57 partitions. Thus, each interval is formed using this *l* parameter;

u_1 = [4600, 4700]

u_2 = [4700, 4800]

…

u_{57} = [10200, 10300]

Step II. Defining fuzzy sets. After partitioning U and finding intervals, linguistic terms for each interval should be defined. These terms ($A_1, A_2, \ldots, A_n$) are represented as fuzzy sets and can be defined by using Eq.8. Thus, each linguistic term is evaluated as follows:

$$A_1 = 1/u_1 + 0.5/u_2 + 0/u_3 + \cdots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n$$

$$A_2 = 0.5/u_1 + 1/u_2 + 0.5/u_3 + \cdots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n$$

$$A_3 = 0/u_1 + 0.5/u_2 + 1/u_3 + \cdots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n$$

.

.

.

$$A_n = 0/u_1 + 0/u_2 + 0/u_3 + \cdots + 0/u_{n-2} + 0.5/u_{n-1} + 1/u_n$$

(16)

The degree of membership of each observation in original time series values belonging to each linguistic term is now defined. Thus, each interval correspondence to all linguistic terms is calculated, for example, for interval u_1 the degree of correspondence to linguistic term A_1 is equal to 1 whereas for A_2 it is equal to 0.5. Similarly, for u_2 , the degree of membership values to A_1, A_2 and A_3 are 0.5, 1 and 0.5 respectively.

Step III. Fuzzification. Now, each observation in original time series data is fuzzified by obtaining its membership degree to each linguistic term (A_1, A_2,.., A_n).

Step IV. Identification of FLRs. FLRs are identified on fuzzified time series by using Definition 2.3. Most studies in the literature uses first-order FLRs (Bose and Mali 2019, Huarng and Yu 2006, Panigrahi and Behera 2020, Singh 2017) but using high-order FLRs can increase the accuracy of forecasting (Cagdas H Aladag *et al.* 2009, Aladag *et al.* 2010b, Kocak *et al.* 2021, Lee *et al.* 2008, Singh 2017) . The identifying of FLRs is very crucial for accuracy of forecasting model and in the literature there are various numbers of studies using different methodologies; statistical approaches, fuzzy relation tables and matrices, and recently machine learning based methodologies (Yu and Huarng 2010, Panigrahi and Behera 2020).

Step V. Defuzzification In the study (Chen 1996), defuzzification step is constructed using following rules:

- If the FLR is empty ($A_i \rightarrow \varnothing$) , then defuzzified forecast calculated as centroid of the corresponding interval u_i .
- If FLR has only one interval in RHS ($A_i \rightarrow A_j$) then defuzzified forecast is calculated as centroid of the corresponding interval u_j.
- If RHS of FLR has more than one intervals ($A_i \rightarrow A_{j1}, A_{j2}, \ldots, A_{jn}$) , then defuzzified forecast is arithmetic mean of corresponding interval centroids:

$$Forecasted_{value} = \left[\frac{A_{j1} + A_{j2} + \cdots + A_{jn}}{n}\right] \quad (17)$$

The study in (Song and Chissom 1993) defines defuzzification rules as follows: (1) if there is only one output membership value (which can be refereed as single maximum value) then forecasted value is the centroid of the interval corresponding to the maximum. (2) if there is more than maximum membership then the arithmetic average is calculated for corresponding intervals.

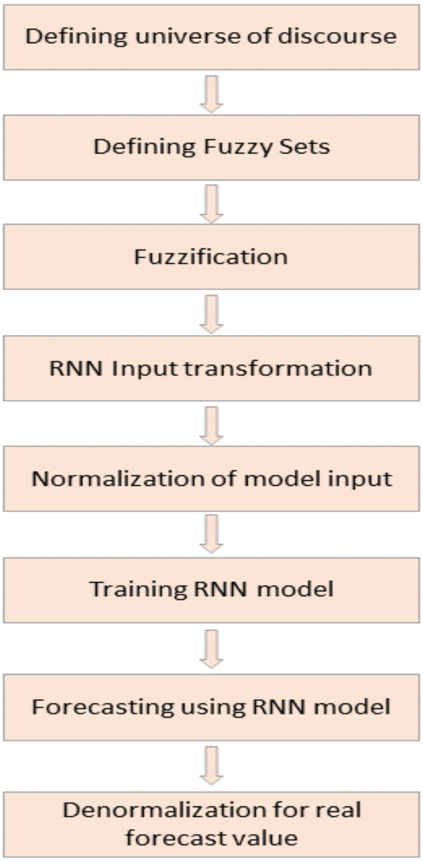

Fig. 6. FTS-GRU forecasting model steps

2.5. Proposed hybrid model.

In FTS domain, neural networks are used in different stages of forecasting. Some of these works have used neural networks in order to define and partition the universe of discourse (Bahrepour *et al.* 2011, Singh and Borah 2014). In (Singh and Borah 2013), a neural network is used in the defuzzification of the FTS values. Moreover, there have been a various number of studies that use a neural network to identify FLRs. Most of these works have used the index number of the interval as input to produce the index number as output (Cagdas H Aladag *et al.* 2009, Aladag *et al.* 2010a, Egrioglu *et al.* 2009, Egrioglu *et al.* 2013, Huarng and Yu 2006). The studies by (Yu and Huarng 2010) and (Kocak *et al.* 2021) have used a neural network model with membership degrees as input/output to determine fuzzy relations.

In this study, we used GRU to identify FLRs (FTS-GRU). The GRU model is applied to the fourth step of FTS modeling (Step IV. Identifying FLRs), and the methodology chart is illustrated in Fig. 6. The proposed model uses membership values obtained in the fuzzification step and also original time series value in order to produce the actual forecast. Thus, our model does not need to defuzzification step. The algorithm for using GRU in FTS is explained in Algorithm 1.

Furthermore, we have used another RNN model (Bidirectional LSTM) as an alternative method for identifying FLRs (FTS-BiLSTM). The algorithm of this model is similar to Algorithm 1 except using two BiLSTM instead of GRU.

The proposed model can handle first-order fuzzy relations and high-order ones since the model input is formed by membership values and also actual values for corresponding original time series. For the first-order FLRs, there is only one real value as additional input to the GRU model. In contrast, for the n-order FLRs, n actual values are used as input to produce a single output, the actual forecast value, as shown in Table **1**.

We can use Fuzzy C-means based partitioning (Li, Cheng, and Lin 2008) or partitioning method proposed by Huarng in (Huarng 2001a) at line 2 of Algorithm1. However, for simplicity, we have used equal length partitioning method. For FTS-BiLSTM, the model implementation at line 11 of the algorithm is replaced with BiLSTM implementation. Both LSTM and GRU models are implemented using *Tensorflow* (Martín Abadi, Ashish Agarwal, Paul Barham *et al.* n.d.).

Algorithm 1: FTS-GRU Methodology

Input: $Y_t(y_1, y_2, \ldots, y_j)$ Original time series

n: number of partitions

Output: acc : accuracy of forecasting

//Define the universe of discourse *U*

$U = [\min(Y_t), \max(Y_t)]$

//Partition *U* by using even length grid partitioning

$\{u_1, u_2, \ldots, u_n\} = [\min(Y_t), \max(Y_t)]/n$

//Compute the midpoints of the intervals

$\forall u \in U, m_i = [u_{i-lower} + u_{i-upper}]/2$

//Fuzzify using membership function

$A\,[j, n] = 0$

$A = f_A(u_1)/u_1 + f_A(u_2)/u_2 + \cdots + f_A(u_n)/u_n$

$m = order\ of\ A$ // define the order of FTS

// Add real time series value to fuzzified time series

for *i = [1, m]*

    $A\,[j, n + i] = (y_{j-m+i})$

//normalize fuzzified time series

$\tilde{A} = normalize(A)$

//split train and test data

$\mathrm{X_{train}}, \mathrm{y_{train}}, \mathrm{X_{test}}, \mathrm{y_{test}} = split(\tilde{A}, 0.8)$

//train GRU model using test data and make prediction

GRU_model = train($\mathrm{X_{train}}, \mathrm{y_{train}}$*)*

$\hat{y}$ = *GRU_model.prediction(*$\mathrm{X_{test}}$)

//Compute accuracy using $\hat{y}$ and $\mathrm{y_{test}}$

acc = *rmse* ($\hat{y}$, $\mathrm{Y_{test}}$)

return acc

Table 1. Sample input for GRU using high-order FLRs(n=2)

A_1	A_2	A_3	A_4	A_5	$Y(t-2)$	$Y(t-1)$	$Y(t)$
0	0.54	1	0.46	0	2720	3126	3522

2.6. Model evaluation

The actual time series values and our model's prediction results are compared to evaluate the proposed work's performance. Two performance metrics, namely Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), are used for this evaluation. These metrics are computed using the following equations;

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Y'_i - Y_i)^2}{n}} \quad (18)$$

$$MAE = \frac{1}{n}\, x\, \sum_{i=1}^{n} |(Y_i - Y'_i)| \quad (19)$$

Table 3. Sample data from TAIEX2000 Dataset.

Date (MM/DD)	Opening Price	Highest Price	Lowest Price	Closing Price
01/04	8,644.91	8,803.61	8,642.50	8,756.55
01/05	8,690.60	8,867.68	8,668.02	8,849.87
01/06	8,876.59	9,023.99	8,863.91	8,922.03
.....	......	......	......	......
10/27	5,991.83	6,003.38	5,805.17	5,805.17
10/30	5,644.26	5,666.96	5,615.90	5,659.08
10/31	5,530.80	5,626.03	5,502.67	5,544.18

3. **Results and Discussion**

To demonstrate proposed model performance, we have conducted accuracy tests on four different datasets (Table 2). The sample from these datasets are shown in Table 3-5.

Table 2. Dataset description

Dataset	Number of observations	
	Train	Test
TAIEX2000	194	48
TAIEX2001	191	48
TAIEX2002	196	49
TAIEX2003	197	49
TAIEX2004	198	49
NIKKEI225	1466	244
COVID-19 CASES	233	58
BIST100 INDEX	3200	800

Table 4. Sample data from NIKKEI225 Dataset.

Date YY/MM/DD	Opening Price	Highest Price	Lowest Price	Closing Price
2011/01/04	10352.1	10409.1	10321.2	10398.0
2011/01/05	10387.9	10413.4	10358.0	10380.7
2011/01/06	10477.5	10530.1	10477.5	10529.7
.....	......	......	......	......
2017/12/20	22834.9	22923.5	22806.7	22891.7
2017/12/21	22852.0	22894.9	22728.0	22866.0
2017/12/22	22850.7	22908.8	22801.1	22902.7

Table 5. Sample data from BIST100 Dataset.

Date YY/MM/DD	Opening Price	Highest Price	Lowest Price	Closing Price
2013/09/19	749.22	751.65	744.99	746.60
2013/09/18	782.14	805.70	782.14	794.66
2013/09/20	794.68	795.06	775.82	778.63

Table 6. Descriptive statistics for BIST100 Dataset.

Mean	Std. Deviation	Min	Max
824.55	186.55	595.67	1133.56

Table 7. Related fuzzy forecasting methods.

Study	Establishing FLR for forecasting
Chen (Chen 1996)	Rule based
Huarng (Huarng and Yu 2006)	Neural network
Aladag (Cagdas H. Aladag et al. 2009)	Neural network
Yu (Yu and Huarng 2010)	Neural network
Aladag (Aladag 2013)	multiplicative neuron model
Bas (Bas et al. 2018)	pi-sigma neural network
Panigrahi (Panigrahi and Behera 2020)	SVM, LSTM and neural network

First of all, in order to show performance comparison of proposed models, Covid-19 active cases of Turkey are used. Dataset contains 291 days of active cases. The RMSE and MAE results are demonstrated in Figure 7. One can observe from Figure 7 that FTS-GRU has better accuracy results than both LSTM based approaches (FTS-BiLSTM and FTS-LSTM). The models are evaluated using first order FLRs. Number of units in hidden layers of each model is set to 32 partition number is set to 40 for this performance test. Moreover, we have tested our approach using BIST100 Index dataset. This dataset contains 4000 values of Turkish Stock Market between 08/06/2005 and 20/09/2013. The descriptive analyisis of this dataset is shown in Table 6.

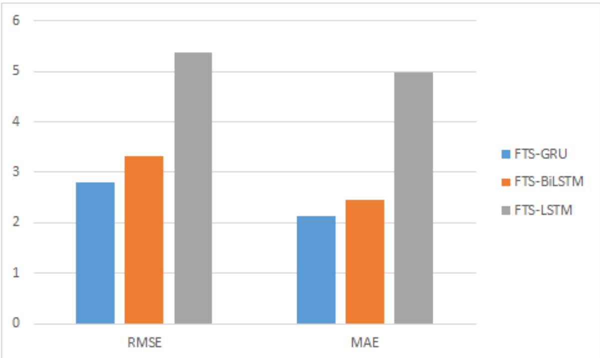

Fig. 7. RMSE and MAE graphic for Covid-19 dataset.

Table 8. RMSE results for TAIEX dataset.

Dataset	Chen (Chen 1996)	Huarng (Huarng and Yu 2006)	Aladag (Cagdas H. Aladag et al. 2009)	Yu (Yu and Huarng 2010)	Aladag (Aladag 2013)	Bas (Bas et al. 2018)	Panigrahi (Panigrahi and Behera 2020)	FTS-GRU	FTS-BiLSTM	FTS-LSTM
TAIEX2000	176.32	152.00	2857.58	**149.59**	2297.60	2297.85	264.00	188.65	201.54	255.69
TAIEX2001	147.84	130.00	564.14	**98.91**	546.40	481.25	173.54	103.57	114.52	278.74
TAIEX2002	101.18	84.00	611.03	**78.71**	530.59	533.68	86.93	155.89	135.60	185.41
TAIEX2003	74.46	**56.00**	644.44	58.78	511.03	142.06	78.36	69.45	98.41	112.51
TAIEX2004	84.28	NA	303.66	55.91	336.47	364.4	73.77	**52.77**	87.13	158.47

Table 9. Test results for Nikkei225 dataset.

Metric	Aladag (Aladag 2013)	Kocak (Kocak et al. 2021)	FTS-GRU (first order)	FTS-GRU (second order)
RSME	58.03	34.47	87.74	51.57
MAE	44.27	27.88	63.57	44.32

Moreover, to show effectiveness of the proposed model we used another dataset, Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) which is widely used in FTS studies (Bose and Mali 2019).

The RMSE results are compared with state-of-art studies in the literature (summarized in Table 7) by taking their RMSE values obtained from (Bose and Mali 2019; Huarng and Yu 2006; Panigrahi and Behera 2020; Singh 2017) and summarized in Table 8. Again, the proposed models are evaluated using first order FLRs. Number of units in hidden layers of each model for each TAIEX dataset are set to (32,32,64,50,64) respectively. Partition number is set to 40 for each TAIEX dataset.

The test results in Table 8 show that the proposed FTS-GRU model outperforms LSTM based approaches again. Moreover, we can observe from the table that FTS-GRU outperforms most of the studies except Yu (Yu and Huarng 2010) but our study shows very close performance. Furthermore, for TAIEX2004 dataset, FTS-GRU has better RMSE values than each of these models.

A recent study of Kocak *et al.* 2021, which uses LSTM for intuitionistic fuzzy time series forecasting applied Nikkei 225 stock exchange dataset for performance evaluation. Thus, we conducted same RMSE performance test by using this dataset and compared our results with the results obtained from (Kocak *et al.* 2021 and Aladag 2013). Moreover, since they have used intuitionistic fuzzy sets, they tried to forecast FTS by using high order FLRs.

Hence, we have evaluated performance comparison by defining first order and also second order FLRs and result are shown in Table 9. For this evaluation, number of hidden layer is set to 8 (eight) and partition number is set to 10 (ten). The test results show that our work outperforms Aladag 2013 but not so good as Kocak *et al.* 2021.

Table 10. Test results for BIST100 dataset.

Metric	Gocken (Gocken et al. 2016)	FTS-GRU (first order)	FTS-GRU (second order)
RSME	2.413	2.207	1.781
MAE	2.813	1.715	1.511

However, the overall RMSE accuracy values are not presented in Kocak *et al.* 2021, rather, they present only a subset of their test set containing 10(ten) elements and its accuracy value. Thus, for a meaningful comparison, the overall performance values for whole dataset should be computed and compared.

The test results for a recent dataset (BIST100 Index) is shown in Table 10. The proposed work is compared with Gocken *et al.* 2016, which integrates metaheuristics and artificial neural networks. The test results also confirm the previous results and our appraoch has better RSME and MAE values for this dataset.

4. Conclusion

Deep learning approaches are extensively used in time series forecasting. However, a few studies use deep learning methods in fuzzy time series forecasting problems in the literature. In this study, we proposed a novel fuzzy time series forecasting model which uses recurrent neural networks. Our model is based on Gated Recurrent Unit (GRU), which has recently been widely used in time series forecasting. The proposed model(FTS-GRU) tries to combine the effectiveness of GRU in nonlinear time series forecasting with the fuzzy set theorem in order to handle uncertainty.

The proposed model evaluates both the time series non-linearity and the inherent uncertainty and ambiguity of the data. At the same time, this model has high accuracy compared to state-art techniques in time series prediction resulting that this model can play an essential role in real-world applications.

Four different data sets are used in the experiments. The model is compared with the Bidirectional LSTM and also a single LSTM-based FTS forecasting model and has better prediction results.

Our numerical experiments revealed that fuzzy time series and deep learning may efficiently be adapted to develop strong, stable, and reliable forecasting models. It is worth mentioning that due to the sensitivity of various hyper-parameters of the proposed models and their high complexity, it is possible that their prediction ability could be further improved by performing additional optimized configuration and mostly feature engineering. Nevertheless, the accuracy results show that our method outperforms most of state-art techniques. Thus the proposed hybrid method can be competitive among these techniques.

Another future work may be to study whether the network model proposed in this paper has the effect of further improving the prediction accuracy on the problem of fuzzy cluster number optimization and hyperparameter selection.

5. References

Aladag, C. H., Basaran M.A., Egrioglu, E., Yolcu U., and Uslu, V.R,. 2009. Forecasting in High Order Fuzzy Times Series by Using Neural Networks to Define Fuzzy Relations. *Expert Systems with Applications,* **36(3, Part 1)**,4228–4231.

Aladag, C. H., 2013. Using Multiplicative Neuron Model to Establish Fuzzy Logic Relationships. *Expert Systems with Applications*, **40(3)**, 850–53.

Aladag, Cagdas Hakan, Ufuk Yolcu, and Erol Egrioglu. 2010a. A High Order Fuzzy Time Series Forecasting Model Based on Adaptive Expectation and Artificial Neural Networks. *Mathematics and Computers in Simulation,* **81(4)**, 875–82.

Aladag, C. H., Yolcu, U., and Egrioglu, E., 2010. A High Order Fuzzy Time Series Forecasting Model Based on Adaptive Expectation and Artificial Neural Networks. *Mathematics and Computers in Simulation,* **81(4)**,875–82.

Aladag, C. H., Yolcu, U., Egrioglu, E., and Dalar A. Z., 2012. A New Time Invariant Fuzzy Time Series Forecasting Method Based on Particle Swarm Optimization. *Applied Soft Computing,* **12(10)**,3291–99.

Arslan S., 2022. A hybrid forecasting model using LSTM and Prophet for energy consumption with decomposition of time series data. *PeerJ Computer Science* **8:e1001**

Bahrepour, M., Akbarzadeh-T. M., Yaghoobi, M., and Naghibi-S., M., B., 2011. An Adaptive Ordered Fuzzy Time Series with Application to FOREX. *Expert Systems with Applications*,**38(1)**,475–85.

Bas, E., Egrioglu, E., Aladag, C. H., and Yolcu, U., 2015. Fuzzy-Time-Series Network Used to Forecast Linear and Nonlinear Time Series. *Applied Intelligence,* **43(2)**,343–55.

Bas, E., Grosan, C., Egrioglu, E., and Yolcu, U., 2018. High Order Fuzzy Time Series Method Based on Pi-Sigma Neural Network. *Engineering Applications of Artificial Intelligence*, **72**,350–56.

Becerra-Rico, J., Aceves-Fernández, M. A., Esquivel-Escalante, K., and Pedraza-Ortega, J. C., 2020. Airborne Particle Pollution Predictive Model Using Gated Recurrent Unit (GRU) Deep Neural Networks. *Earth Science Informatics,* **13(3)**, 821–34.

Bose, M., and Mali K., 2019. Designing Fuzzy Time Series Forecasting Models: A Survey. *International Journal of Approximate Reasoning,* **111,** 78–99.

Bulut, E. 2014. Modeling Seasonality Using the Fuzzy Integrated Logical Forecasting (FILF) Approach. *Expert Systems with Applications,* **41(4, Part 2),** 1806–12.

Cagcag Yolcu, O., and Lam, H. K., 2017. A Combined Robust Fuzzy Time Series Method for Prediction of Time Series. *Neurocomputing* **247**, 87–101.

Cai, Q., Zhang, D., Zheng, W., and Leung, S. C. H., 2015. A New Fuzzy Time Series Forecasting Model Combined with Ant Colony Optimization and Auto-Regression, *Knowledge-Based Systems,* **74,** 61–68.

Castillo, O., Alanis, A., Garcia, M., and Arias, H., 2007. An Intuitionistic Fuzzy System for Time Series Analysis in Plant Monitoring and Diagnosis. *Applied Soft Computing*,**7(4),** 1227–33.

Chang, J. R., Liang-Ying W., and Cheng, C. H., 2011. A Hybrid ANFIS Model Based on AR and Volatility for TAIEX Forecasting. *Applied Soft Computing,* **11(1),** 1388–95.

Chen, M. Y., and Chen, B. T., 2014. Online Fuzzy Time Series Analysis Based on Entropy Discretization and a Fast Fourier Transform. *Applied Soft Computing,* **14,** 156–66.

Chen, S. M., and Kao, P. K., 2013. TAIEX Forecasting Based on Fuzzy Time Series, Particle Swarm Optimization Techniques and Support Vector Machines. *Information Sciences,* **247,** 62–71.

Chen, S. M., and Tanuwijaya, K., 2011. Fuzzy Forecasting Based on High-Order Fuzzy Logical Relationships and Automatic Clustering Techniques. *Expert Systems with Applications,* **38(12),** 15425–37.

Chen, S. M., 1996. Forecasting Enrollments Based on Fuzzy Time Series. *Fuzzy Sets and Systems,* **81(3)**, 311–19.

Cho, K., Van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y., 2014. Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation. BT - Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A Meeting of SIGDAT,. 1724–1734.

Cui, Z., and Wang, Y., 2017. Deep Stacked Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-Wide Traffic Speed Prediction.

Dutta, A., Kumar, S., and Basu, M., 2020. A Gated Recurrent Unit Approach to Bitcoin Price Prediction. *Journal of Risk and Financial Management* **13(2)**.

Efendi, R., Ismail, Z., and Deris, M. M., 2015. A New Linguistic Out-Sample Approach of Fuzzy Time Series for Daily Forecasting of Malaysian Electricity Load Demand. *Applied Soft Computing*, **28**, 422–30.

Egrioglu, E., Aladag, C. H., and Yolcu, U., 2013. Fuzzy Time Series Forecasting with a Novel Hybrid Approach Combining Fuzzy C-Means and Neural Networks. *Expert Systems with Applications*, **40(3)**, 854–57.

Egrioglu, E., Aladag, C. H., Yolcu, U., Uslu, V. R., and Basaran, M. A., 2009. A New Approach Based on Artificial Neural Networks for High Order Multivariate Fuzzy Time Series. *Expert Systems with Applications,* **36(7),** 10589–94.

George E. P. Box, Jenkins, G. M., Reinsel, G. C., Ljung, G. M., 2015. *Time Series Analysis: Forecasting and Control*. John Wiley & Sons.

De Gooijer, J. G., and Hyndman, R. J., 2006. 25 Years of Time Series Forecasting. *International Journal of Forecasting,* **22(3),** 443–73.

Gocken, M., Ozcalici M., Boru A., and Dosdogru A.T., 2016. Integrating metaheuristics and Artificial Neural Networks for improved stock price prediction. *Expert Syst. Appl.*, **44**, pp. 320–331.

Graves, A., Mohamed, A., and Hinton, G. E., 2013. Speech Recognition with Deep Recurrent Neural Networks. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 6645-6649.

Hájek, P., and Olej, V., 2017. Intuitionistic Neuro-Fuzzy Network with Evolutionary Adaptation. *Evolving Systems,* **8(1),** 35–47.

Huarng, K., 2001. Effective Lengths of Intervals to Improve Forecasting in Fuzzy Time Series. *Fuzzy Sets and Systems,* **123(3),** 387–394.

Huarng, K., 2001. Heuristic Models of Fuzzy Time Series for Forecasting. *Fuzzy Sets and Systems* **123(3)**, 369–386.

Huarng, K., and Yu, T. H. K., 2006. The Application of Neural Networks to Forecast Fuzzy Time Series. *Physica A: Statistical Mechanics and Its Applications,* **363(2),** 481–491.

Jang, J. R. 1993. ANFIS: Adaptive-Network-Based Fuzzy Inference System. *IEEE Transactions on Systems, Man, and Cybernetics,* **23(3),** 665–85.

Jozefowicz, R., Zaremba, W., and Sutskever, I., 2015. An Empirical Exploration of Recurrent Network Architectures. *32nd International Conference on Machine Learning, ICML 2015* **3**, 2332–40.

Kocak, C., 2017. ARMA(p,q) Type High Order Fuzzy Time Series Forecast Method Based on Fuzzy Logic Relations. *Applied Soft Computing* **58**, 92–103.

Kocak, C., Egrioglu, E., and Bas, E., 2021. A New Deep Intuitionistic Fuzzy Time Series Forecasting Method Based on Long Short-Term Memory. *Journal of Supercomputing,* **77(6)**, 6178–96.

Lee, L. W., Wang, L. H., and Chen, S. M., 2008. Temperature Prediction and TAIFEX Forecasting Based on High-Order Fuzzy Logical Relationships and Genetic Simulated Annealing Techniques. *Expert Systems with Applications,* **34(1)**, 328–36.

Li, S. T., Cheng, Y. C., and Lin, S. Y., 2008. A FCM-Based Deterministic Forecasting Model for Fuzzy Time Series. *Computers & Mathematics with Applications,* **56(12)**, 3052–63.

Lin, T., Horne, B. G., and Giles, C. L., 1998. How Embedded Memory in Recurrent Neural Network Architectures Helps Learning Long-Term Temporal Dependencies, **11(5)**, 861–68.

Lindemann, B., Müller, T., Vietz, H., Jazdi, N., and Weyrich, M., 2021. A Survey on Long Short-Term Memory Networks for Time Series Prediction. *Procedia CIRP,* **99**, 650–55.

Martín A., Agarwal, A., Barham, P., Brevdo, E., Chen,A. D. Z., Citro, C., Corrado,G. S., Goodfellow, I., Dean, J., Devin, M., Ghemawat, S., Jia, Y., Harp, A., Irving, G., Isard, M., Jozefowicz, R., and Yuan Yu, X. Z., n.d. TensorFlow:Large-Scale Machine Learning on Heterogeneous Systems. Retrieved (https://www.tensorflow.org/) (July 2022).

Nie, J.. 1997. Nonlinear Time-Series Forecasting: A Fuzzy-Neural Approach. *Neurocomputing*, **16(1),** 63–76.

Novák, V.. 1995. Towards Formalized Integrated Theory of Fuzzy Logic BT - Fuzzy Logic and Its Applications to Engineering, Information Sciences, and Intelligent Systems. 353–63 in, edited by Z. Bien and K. C. Min. Dordrecht: Springer Netherlands.

Panigrahi, S., and Behera, H. S., 2020. A Study on Leading Machine Learning Techniques for High Order Fuzzy Time Series Forecasting. *Engineering Applications of Artificial Intelligence*, **87**.

Sadaei, H. J., Enayatifar,R., Guimarães, F. G., Mahmud, M., and Alzamil, Z. A., 2016. Combining ARFIMA Models and Fuzzy Time Series for the Forecast of Long Memory Time Series, **175,** 782–796.

Sadaei, H. J., Silva, P. D. L., Guimarães, F. G., and Lee, M. H., 2019. Short-Term Load Forecasting by Using a Combined Method of Convolutional Neural Networks and Fuzzy Time Series. *Energy,* **175,** 365–377.

Schuster, M., and Paliwal, K. K., 1997. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing,* **45(11),** 2673–2681.

Shen, G., Tan, Q., Zhang, H., Zeng, P., and Xu, J. 2018. Deep Learning with Gated Recurrent

Unit Networks for Financial Sequence Predictions. *Procedia Computer Science,* **131**, 895–903.

Siami-Namini, S., Tavakoli,N., and Namin, A. S., 2019. The Performance of LSTM and BiLSTM in Forecasting Time Series. Pp. 3285–92 in *2019 IEEE International Conference on Big Data (Big Data)*.

Singh, P., 2017. A Brief Review of Modeling Approaches Based on Fuzzy Time Series. *International Journal of Machine Learning and Cybernetics,* **8(2),** 397–420.

Singh, P., and Borah, B., 2013. High-Order Fuzzy-Neuro Expert System for Time Series Forecasting. *Knowledge-Based Systems*, **46**, 12–21.

Singh, P., and Borah, B., 2014. An Effective Neural Network and Fuzzy Time Series-Based Hybridized Model to Handle Forecasting Problems of Two Factors. *Knowledge and Information Systems*, **38(3),** 669–90.

Song, Q., and Chissom, B. S., 1993. Fuzzy Time Series and Its Models. *Fuzzy Sets and Systems* **54(3)**, 269–77.

Stefanakos, C., 2016. Fuzzy Time Series Forecasting of Nonstationary Wind and Wave Data. *Ocean Engineering*, **121**, 1–12.

Sullivan, J., and Woodall, W. H., 1994. A Comparison of Fuzzy Forecasting and Markov Modeling. *Fuzzy Sets and Systems,* **64(3)**, 279–93.

Tan, Q., Mang Y., Baoyao Y., Siqi L., Ma, A. J., Cheuk-Fung Yip, T., Wong, G. L. H., and Yuen, P. C., 2020. DATA-GRU: Dual-Attention Time-Aware Gated Recurrent Unit for Irregular Multivariate Time Series. *Proceedings of the AAAI Conference on Artificial Intelligence* , **34(01)**, 930–37.

Torbat, S., Khashei, M., and Bijari, M., 2018. A Hybrid Probabilistic Fuzzy ARIMA Model for Consumption Forecasting in Commodity Markets. *Economic Analysis and Policy* **58**, 22–31.

Tran, N., Nguyen, N., Nguyen, B. M., and Nguyen, G., 2018. A Multivariate Fuzzy Time Series Resource Forecast Model for Clouds Using LSTM and Data Correlation Analysis. *Procedia Computer Science* ,**126** ,636–45.

Tsaur, R. C., Yang, J. C. O., and Wang, H. F., 2005. Fuzzy Relation Analysis in Fuzzy Time Series Model. *Computers & Mathematics with Applications,* **49(4)**, 539–48.

Tseng, F. M., Tzeng, G. H., Yu, H. C., and Yuan, B. J. C., 2001. Fuzzy ARIMA Model for Forecasting the Foreign Exchange Market. *Fuzzy Sets and Systems,* **118(1)**, 9–19.

Wang, Y, Wenlong L., and Chang, Y., 2018. Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting. *Energies* 11(8).

Wong, H. L., Tu, Y. H., and Wang, C. C., 2010. Application of Fuzzy Time Series Models for Forecasting the Amount of Taiwan Export. *Expert Systems with Applications,* **37(2),** 1465–1470.

Yang, S., Yu, X., Zhou, Y., and Yu, X., 2020. LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example. Pp. 98–101 in *Proceedings - 2020 International Workshop on Electronic Communication and Artificial Intelligence, IWECAI 2020*.

Yu, T. H. K., and Huarng, K. H., 2010. A Neural Network-Based Fuzzy Time Series Model to Improve Forecasting. *Expert Systems with Applications,* **37(4)**, 3366–72.

Zadeh, L. A., 1965. Fuzzy Sets. *Information and Control,* **8(3)**, 338–53.