INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 16, Issue: 4, Month: August, Year: 2021
Article number: 3846, doi.org/10.15837/ijccc.2021.4.3846

CCC Publications
AGORA UNIVERSITY PRESS

The Modeling of Interval-Valued Time Series Using Possibility Measure-Based Encoding-Decoding Mechanism

Zefeng Lv, Dan Shan, Xiaopeng Hu, Wei Lu

Zefeng Lv
School of Computer Science and Technology
Dalian University of Technology
Dalian, 116023 China
kkzf_2001@163.com

Dan Shan
Department of Electronic Engineering
Dalian Neusoft University of Information
Dalian, 116023 China
shandan@neusoft.edu.cn

Xiaopeng Hu
School of Computer Science and Technology
Dalian University of Technology
Dalian, 116023 China
xphu@dlut.edu.cn

Wei Lu*
School of Control Science and Engineering
Dalian University of Technology
Dalian, 116023 China
*Corresponding author: luwei@dlut.edu.cn

Abstract

Interval-valued time series (ITS) is a collection of interval-valued data whose entires are ordered by time. The modeling of ITS is an ongoing issue pursued by many researchers. There are diverse ITS models showing better performance. This paper proposes a new ITS model using possibility measure-based encoding-decoding mechanism involved in fuzzy theory. The proposed model consists of four modules, say, linguistic variable generation module, encoding module, inference module and decoding module. The linguistic variable generation module can provide a series of linguistic variables expressed in fuzzy sets used to described dynamic characteristics of ITS. The encoding module encodes ITS into some embedding vectors with semantics with the aid of possibility measure and linguistic variables formed by linguistic variable generation module. The inference module uses artificial neural network to capture relationship implied in those embedding vectors with semantic. The decoding module decodes for the outputs of the inference module to produce the output of linguistic and interval formats by using the possibility measure-based encoding-decoding mechanism. In comparison with existing ITS models, the proposed model can not only produce the output of linguistic format, but also exhibit better numeric performance.

Keywords: interval-valued time series, granular computing, information granules, fuzzy sets, possibility measure.

1 Introductory comments

Interval-valued time series (ITS) is a collection of interval-valued data whose elements are ordered by time, where time can be replaced by any other variable of some physical meaning. In general, ITS is denoted as $\boldsymbol{X} = \{X_t = [x_t^L, x_t^U] | x_t^L \leq x_t^U; x_t^L, x_t^U \in \mathbb{R}; t = 1, 2, \cdots, k\}$, where $X_t = [x_t^L, x_t^U]$ is an interval number, x_t^L and x_t^U represent the lower and upper bounds of the observed variable X at time t, respectively. ITS coming with interval format can depict the uncertainty and variability of variables, which has been applied into many fields such as finance, energy, environment, agriculture etc. As a description of temporal relationships in interval format, the modeling of ITS is an ongoing topic pursued by many researchers. There are diverse methods devoted to develop ITS models. These methods fall with the following three classes.

The first class is the statistical theory-based methods. In general, this class of methods exploits some existing statistic modeling methods to develop ITS model. When using this class of methods to model, ITS is first decomposed to the interval mid-point series $\boldsymbol{M} = \{M_t = \frac{x_t^U + x_t^L}{2} | t = 1, 2, \cdots, k\}$ and the interval radius series $\boldsymbol{R} = \{R_t = \frac{x_t^U - x_t^L}{2} | t = 1, 2, \cdots, k\}$ by calculating respectively interval mid-points and interval radius. Subsequently, some classic statistical modeling methods [15] like autoregressive (AR), autoregressive moving average (ARMA), and exponential smoothing etc. are invoked to make these two time series model, respectively. As a result, two numeric models concerning those two time series are produced, which are regarded as a whole and become the model of the ITS. When the ITS model is used to carry out prediction, the interval mid-point and interval radius that deliver by the model are recombined to form the lower and upper boundaries of predictive interval, say, $\hat{x}_t^U = M_t + R_t$ and $\hat{x}_t^L = M_t - R_t$, where $\hat{x}_t^U$ and $\hat{x}_t^L$ are the predictive upper lower boundaries. Along this idea, Maia *et al.* [9] presented AR model and ARMA model of ITS, where these two ITS models are denoted as AR^{I} and $\mathrm{ARMA}^{\mathrm{I}}$, respectively. Maia and de Carvalho [10] extended the Holt's exponential smoothing method [7] encountered frequently in numeric time series to make ITS model. The resulting model is termed as the interval Holt model ($\mathrm{Holt}^{\mathrm{I}}$). By extending the asymptotic distribution theory [6] to deal with interval-valued data, Sun and Han *et al.* [16] presented an interval-valued threshold autoregressive models (iTAR) for ITS. Wang and Song *et al.* [19] develop a granular model to compensate piecewise the residual error of iTAR model for enhancing further its performance.

The second class is the blackbox model-based methods, where the blackbox model could be artificial neural network (ANN) [5] or multi-output support vector regression (MSVR) [18]. This class of methods first organize ITS into a series of input-output pairs in the form $\{(x_{t-1}^L, x_{t-1}^U, x_{t-2}^L, x_{t-2}^U, \cdots, x_{t-d}^L, x_{t-d}^U); (x_t^L, x_t^L)\}$, where d is a delay factor. Then the architecture of blackbox are determined in terms of those input-output pairs, and the corresponding parameters are learned in supervised manner. In such way, the blackbox model for ITS is finally produced. Bearing this in mind, Maia *et al.* [9] proposed the ANN-based ITS model, named as $\mathrm{ANN}^{\mathrm{I}}$, where two independent ANNs are used to fit the interval mid-point series $\boldsymbol{M}$ and interval radius series $\boldsymbol{R}$. Maia and de Carvalho [10] also proposed the multilayer perceptrons (MLP)-based ITS model, which is named as $\mathrm{ANN}^{\mathrm{II}}$. Here the MLP includes $2d$ inputs and two outputs. Xiong *et al.* [20] also developed the ANN-based ITS model by considering the interval numbers as the complex numbers and making use of the fully complex-valued radial basis function neural networks [17]. Similarly, Xiong and Bao *et al.* [21] also proposed the MSVR model of ITS (which is named as $\mathrm{MSVR}^{\mathrm{I}}$), where MSVR is used to fit the input-output pairs produced by ITS and the corresponding parameters are learned by firefly algorithm [23]. Since the blackbox model like ANN or support vector regression (SVR) etc. has the well ability to approximate any nonlinear function, this class of methods can deal with the non-stationary and nonlinear problems encountered in ITS.

The last class is the hybrid methods that combine with the above-mentioned two class of ones. Maia *et al.* [9] combines the above $\mathrm{ARIMA}^{\mathrm{I}}$ and $\mathrm{ANN}^{\mathrm{II}}$ models into the hybrid model $\mathrm{ARIMA}^{\mathrm{I}} - \mathrm{ANN}^{\mathrm{II}}$. Maia and de Carvalho [10] integrates the above $\mathrm{Holt}^{\mathrm{I}}$ and $\mathrm{ANN}^{\mathrm{II}}$ models into the hybrid model $\mathrm{Holt}^{\mathrm{I}} - \mathrm{ANN}^{\mathrm{II}}$. Xiong and Li *et al.* [22] also designed a hybrid model of ITS, named as $\mathrm{Holt}^{\mathrm{I}} - \mathrm{MSVR}^{\mathrm{I}}$, by combining $\mathrm{Holt}^{\mathrm{I}}$ and $\mathrm{MSVR}^{\mathrm{I}}$ models. The ITS models established by the hybrid methods exhibit better performance in comparison with the above two class of ones.

Albeit the above-mentioned statistical theory-, blackbox model-, and hybrid method-based ITS model have exhibited better performance in many fields, they exist also several evident limitations. One is that the above-mentioned methods are all failed when ITS includes linguistic variables such as "higher amplitude", "low amplitude" and so on. Another is that most of the existing methods require to decompose ITS into interval mid-point and interval radius time series, which increases additional computation. In addition, the above methods do not consider the interpretability of output of ITS model, which means that the output of ITS models established by the above methods does not have semantics. To reduce these limitations, Lu *et al.* [8] proposed the linguistic modeling framework of ITS based on granular computing (GrC) [11]. In the framework, ITS are first regarded as information granules expressed in some formalism and encoded by a predefined granular codebook. Then the encoded results are delivered the multilayer perceptrons based granular inference system to carry out causal inference. Finally, the inferred results are decoded by the predefined granular codebook to produced the interval-valued output with linguistic landmark. In this framework, Zhou *et al.* [24] proposed the granular model of ITS, where interval is regarded as an information granule expressed in fuzzy set.

Inspired by GrC idea, in this study, a novel hybrid ITS model where the encoding-decoding scheme based on possibility measure [3] is introduced into the above granular modeling framework of ITS is proposed. The model is consisted of four modules. The first module is linguistic variable generation module, which can produce some linguistic variables expressed in the form of fuzzy sets. These linguistic variables depict characteristics of ITS. The second module is the encoding module, which focuses on making ITS semantics. In this module, ITS is taken as a series of information granules expressed in interval format, and is encoded into some embedding vectors with semantics in terms of linguistic variables formed already in the first module by the possibility measure-based encoding mechanism. Entries in the embedding vector is the value of possibility degree of the interval-valued data versus individual linguistic variables. The third module is inference one in which ANN is exploited to capture relations implying in those embedding vectors with semantics, and to perform causal reasoning. The last module is the decoding one. It decodes for the reasoning results produced by the inference module by means of two different types of possibility measure-based schemes to produce different format of outputs. The proposed modeling method exhibits the following advantages.

- Possibility measure-based encoding-decoding mechanism is introduced into the modeling process of ITS, which improves performance of the ensuing ITS model.
- The proposed modeling method does not increase extra computational consumption since interval number is regarded as a whole (an information granule) and participate into the subsequent modeling process of ITS.
- The proposed model can produce not only the output in linguistic format, but also the output in interval format.

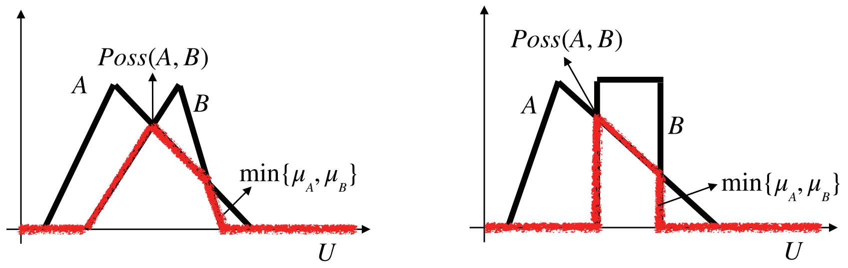
The paper is organized as follows. In Section 2, we introduce the possibility measure-based encoding-decoding mechanism. Section 3 presents the proposed modeling method of ITS in detail. Experiments on several financial datasets from the real world are covered in Section 4. Some discussions concerning the impact of the parameter involved in the proposed modeling method on performance of corresponding ITS model are also involved in this section. Section 5 provides some concludes.

2 Possibility measure-based interval encoding and decoding mechanisms

In this section, possibility measure is first recalled. In a sequel, the possibility measure-based interval encoding and decoding mechanisms are detailed.

2.1 Possibility measure

The possibility measure [3] originates from possibility theory [4]. Possibility theory is an extension of the theory of fuzzy sets and fuzzy logic [25], which is introduced to represent and operate uncertainty



(a) Possibility measure between two fuzzy sets

(b) Possibility measure between fuzzy set and interval

Figure 1: Calculations of possibility measure.

conveyed by natural language. In possibility theory, possibility measure is used to assess the similarity between two fuzzy sets in the same universe of discourse.

Consider two fuzzy sets A and B in the universe of discourse U, their possibility measure is defined as

$$\lambda = Poss(A, B) = \sup_{x \in U} [\min\{\mu_A(x), \mu_B(x)\}], \tag{1}$$

where μ_A and μ_B are membership functions of fuzzy sets A and B, respectively. The definition of above possibility measure can also generalized by admitting any t-norm to replace the minimal operator. In this case, the possibility measure can be rewritten as

$$\lambda = Poss(A, B) = \sup_{x \in U} [A(x)\, t\, B(x)],$$

where the symbol t stands for any t-norm. Computationally, the possibility measure $Poss(A, B)$ can quantify the optimistic degree of overlap between two fuzzy sets A and B. Specifically, the total of n ordered probes with span d is first generated uniformly within the universe of discourse U. These probes are expressed in the form $prob_i \in U$ with $i = 1, 2, \cdots, n$, where $prob_1 < prob_2 < \cdots < prob_n$ and $prob_i - prob_{i-1} = d$ $(i = 2, 3, \cdots, n)$. Subsequently, for each of probes, we calculate the minimum of membership degrees between it with respect to fuzzy sets A and B, say, $\min\{A(prob_i), B(prob_i)\}$, $i = 1, 2, \cdots, n$. Finally, the value of possibility measure between fuzzy sets A and B can be determined by picking up a highest one among those minimums. When the value of d trends to zero, the probability measure in (1) can be solved accurately. Fig.1a visualizes calculation of possibility measure between fuzzy sets A and B. Furthermore, since interval can be regarded as a kind of degenerated fuzzy set, the possibility measure between fuzzy set and interval can also be calculated in light of the above process, see Fig.1b.

2.2 Possibility measure-based interval encoding-decoding mechanism

Using possibility measure-based encoding-decoding mechanism can realize reconstruction of interval. In this subsection, we first formulate the problem of interval reconstruction, and then elaborate on the realization of interval reconstruction by means of possibility measure-based encoding-decoding mechanism.

Suppose that c reference fuzzy sets in the universe of discourse U, denoted as $A_1, A_2, \cdots, A_c$, and an interval $B = [a, b]$ defined in U. The interval B can be encoded by means of reference fuzzy sets in the form of possibility measure, which means that the interval B is encoded into a vector $\langle \lambda_1, \lambda_2, \cdots, \lambda_c \rangle$, of which, λ_i $(i = 1, 2, \cdots, c)$ is obtained by (1) with the aid of a series of probes. The reconstruction problem of interval B is to estimate the lower bound a and upper bounds b of interval B by making use of those well-defined reference fuzzy sets and values of possibility measure.

To address the above reconstruction problem of interval B, we resort to possibility measure-based decoding mechanism that are already presented in the theory of possibility theory [1, 12]. In light of the decoding mechanism, the method of reconstructing interval $B = [a, b]$ is detailed as follows, where the key of reconstruction of interval is to estimate the upper bounds of values of the eigenfunction of interval over the entire universe of discourse.

Specifically, the total of n ordered probes with span d are first generated uniformly in U, say, $prob_1, prob_2, \cdots, prob_n \in U$ with $prob_i - prob_{i-1} = d$ $(i = 2, 3, \cdots, n)$. Subsequently, considering the reference fuzzy set A_j $(j = 1, 2, \cdots, c)$ and the possibility measure value λ_j of the interval to be estimated versus this fuzzy set, we can estimate the upper bounds of values of the eigenfunction of interval B concerning individual probes by the following formula

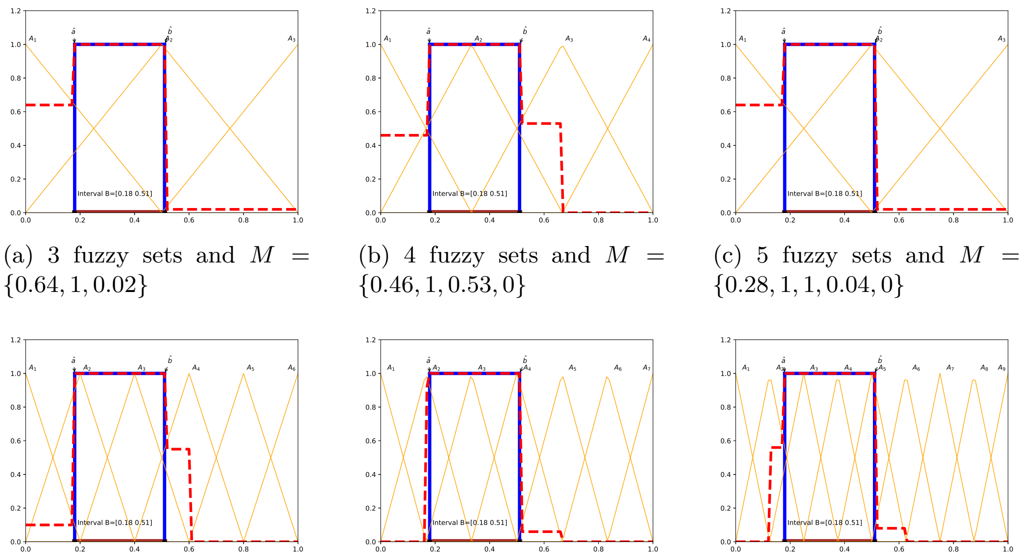
$$\hat{f}_j(x) = A_j(x)\varphi\lambda_j = \begin{cases} 1, & \text{if } A_j(x) \leq \lambda_j \\ \lambda_j, & \text{if } A(x) > \lambda_j \end{cases}, \tag{2}$$

where $x \in \{prob_1, prob_2, \cdots, prob_n\}$. Thus, in the universe of discourse U, a total of n estimated upper bounds of values of the eigenfunction of interval B are generated for all probes with the aid of reference fuzzy set A_j, viz., $\hat{f}_j(p_i)$ with $i = 1, 2, \cdots, n$. In the same way, for other reference fuzzy sets, the estimated upper bounds of values of the eigenfunction of interval B with regard to individual probes can also obtained. Further, those upper bounds of eigenfunction values estimated by reference fuzzy sets can be aggregated by the minimal operator in terms of individual probes, namely,

$$\hat{F}(p_i) = \min\{\hat{f}_1(p_i), \hat{f}_2(p_i), \cdots, \hat{f}_c(p_i)\} \tag{3}$$

with $i = 1, 2, \cdots, n$, which results in the formation of estimated upper bounds of values of the eigenfunction of interval B to be reconstructed. Note that the values of $\hat{F}$ distribute completely over the entire universe of discourse U when the prob span d used to generate probes tends to zero. Once the estimation of upper bounds of values of the eigenfunction of interval B to be reconstructed have been determined, we can select respectively the minimal and maximal points among those probes that make the value of $\hat{F}$ be equal to 1 as the estimations of lower and upper bounds of interval B, i.e., $\hat{B} = [\hat{a}, \hat{b}]$ with $\hat{a} = \min\{prob_i | \hat{F}(prob_i) = 1, i = 1, 2, \cdots, n\}$ and $\hat{b} = \max\{prob_i | \hat{F}(prob_i) = 1, i = 1, 2, \cdots, n\}$. It is important to note that if reference fuzzy sets are triangular fuzzy sets whose membership functions are characterized with $\frac{1}{2}$ overlap degree and the prob span d used to uniformly generate probes tends to zero, the reconstruction of interval B with zero error happens, viz., $\hat{a} = a$ and $\hat{b} = b$.

In what follows, we present an illustrative example to show the reconstruction of interval $[0.18, 0.51]$ by exploiting the possibility measure-based decoding mechanism, where several different number of reference fuzzy sets with triangular membership functions characterized by $\frac{1}{2}$ overlap and different



(a) 3 fuzzy sets and $M = \{0.64, 1, 0.02\}$

(b) 4 fuzzy sets and $M = \{0.46, 1, 0.53, 0\}$

(c) 5 fuzzy sets and $M = \{0.28, 1, 1, 0.04, 0\}$

(d) 6 fuzzy sets and $M = \{0.1, 1, 1, 0.55, 0, 0\}$

(e) 7 fuzzy sets and $M = \{0, 0.92, 1, 1, 0.06, 0, 0\}$

(f) 9 fuzzy sets and $M = \{0.56, 1, 1, 1, 0.08, 0, 0, 0, 0\}$

Figure 2: Upper bound of distribution of estimation of eigenfunction value of interval $[0.18, 0.51]$.

Table 1: The reconstruction of interval $[0.18, 0.51]$ in the case of different numbers of fuzzy sets and different sizes of prob span.

c	Possibility measure (Encoding)									Reconstructed interval (Decoding)		
	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	$d = 0.2$	$d = 0.1$	$d = 0.01$
3	0.64	1.0	0.02	—	—	—	—	—	—	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$
4	0.46	1.0	0.53	0	—	—	—	—	—	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$
5	0.28	1.0	1.0	0.04	0	—	—	—	—	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$
6	0.1	1.0	1.0	0.55	0	0	—	—	—	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$
7	0	0.92	1.0	1.0	0.06	0.0	0.0	—	—	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$
9	0	0.56	1.0	1.0	1.0	0.08	0	0	0	$[0.2, 0.4]$	$[0.2, 0.5]$	$[0.18, 0.51]$

sizes of prob span are used. Tab.1 reports the corresponding results for the reconstruction of interval. Fig.2 also visualizes the reconstruction process of interval for the prob span d being 0.01, where the red dashed line is the estimated upper bounds of eigenfunction values of interval to be reconstructed. From Tab.1, we can observe that the zero reconstruction error happens when the prob span is a smaller value in making use of possibility measure-based decoding scheme with triangular fuzzy sets whose membership functions are characterized by $\frac{1}{2}$ overlap to rebuild an interval.

3 The modeling of ITS using possibility measure-based interval encoding-decoding scheme

In this section, the construction process of the proposed ITS model is first detailed, and then an illustrative example is presented to exhibit the process. In the construction of the model, an ITS including k interval-valued data coming with the form $\boldsymbol{X} = \{X_t = [x_t^L, x_t^U] \mid x_t^L \leq x_t^U, x_t^L, x_t^U \in \mathbb{R}, t = 1, 2, \cdots, k\}$ is considered.

3.1 The construction of proposed ITS model

The proposed modeling method of ITS starts with the determination of corresponding universe of discourse U. we denote $min\{x_1^L, x_2^L, \cdots, x_k^L\}$ and $max\{x_1^U, x_2^U, \cdots, x_k^U\}$ as x_{min} and x_{max}, respectively. Let l_1 and l_2 respectively become the trim factors of lower and upper bounds, where $l_1, l_2 > 0$. Thus the universe of discourse U can be obtained, i.e., $U = [u_1, u_2] = [x_{min} - l_1, x_{max} + l_2]$. The framework of the proposed modeling method of ITS $\boldsymbol{X}$ is vividly presented in Fig.3. It is consisted of the following four modules: 1) linguistic variable generation module, 2) encoding module, 3) inference module and 4) decoding module.

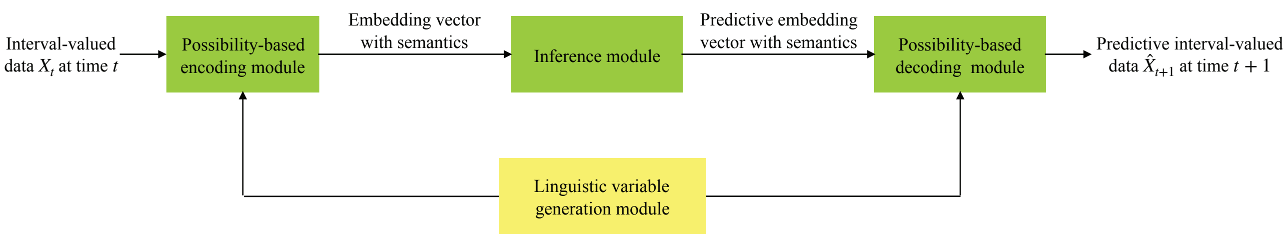


Figure 3: The framework of modeling ITS using possibility measure-based interval encoding-decoding mechanism.

The main function of linguistic variable generation module is to generate c ($c \geq 2$) linguistic variables used to describe amplitude characteristics of ITS. The formed linguistic variables are used to make ITS semantics in a sequel. Since fuzzy sets can be acted as a bridge to link data with natural language, our task in this module is essentially to design c linguistic variables expressed in fuzzy sets coming with some semantics on the discourse of universe U. To design these fuzzy sets, there are two compelling requirements to be simultaneously addressed: 1) the universe of discourse U is covered by these fuzzy sets, and 2) at least one in these fuzzy sets can be activated by any element located in the universe of discourse U. In order to satisfy both these two requirements, here we consider to directly construct c triangular fuzzy sets whose membership functions are distributed uniformly on the the universe of discourse U and characterized by $\frac{1}{2}$ overlap. The reason behind this consideration is that the zero reconstruction error can be produced when using these triangle fuzzy sets to reconstruct interval (see 2.2 for details). Specifically, we first divide the entire universe of discourse $U = [u_1, u_2]$ into $c-1$ segments evenly, which results in the formation of $c-2$ ordered split points. These split points can be denoted as $s_1, s_2, \cdots, s_{c-2}$, where $u_1 < s_1 < s_2 < \cdots < s_{c-2} < u_2$ and $s_i = \frac{iu_2+(c-i-1)u_1}{c-1}$ with $i = 1, 2, \cdots, c-2$. Next these produced split points together with two end points u_1 and u_2 of the universe of discourse U are taken as the kernels of c triangular fuzzy sets to be constructed. The interval composed of the two points adjacent to individual kernels (viz. the points of the left and right hand sides of each kernel) is served as the support of the corresponding triangular fuzzy set, where when the left end point of U, u_1, becomes the kernel of some triangle fuzzy set, the point of the left hand side of this kernel coincides with u_1 whereas when the right end point of U, u_2, becomes the kernel of some triangular fuzzy set, the point of the right hand side of this core coincides with u_2. In this way, the total of c triangular fuzzy sets whose membership functions are characterized by $\frac{1}{2}$ overlap are generated completely, and the corresponding member functions are expressed as follows

$$A_1(x) = \begin{cases} -\frac{x-s_1}{s_1-u_1}, & x \in [u_1, s_1] \\ 0, & otherwise \end{cases}, \cdots, A_i(x) = \begin{cases} \frac{x-s_{i-2}}{s_{i-1}-s_{i-2}}, & x \in [s_{i-2}, s_{i-1}] \\ -\frac{x-s_i}{s_i-s_{i-1}}, & x \in [s_{i-1}, s_i] \\ 0, & otherwise \end{cases}, \cdots,$$
$$A_c(x) = \begin{cases} \frac{x-s_{c-2}}{u_2-s_{c-2}}, & x \in [s_{c-2}, u_2] \\ 0, & otherwise \end{cases}, \tag{4}$$

where $s_0 = u_1$ and $s_{c-1} = u_2$. Once c triangular fuzzy sets with membership functions being $\frac{1}{2}$ overlap have been constructed on the discourse of universe U, we associate respectively linguistic variables like "*amplitude is low*", "*amplitude is medium*", "*amplitude is high*" etc. with the above fuzzy sets in terms of the distribution of their support on the universe of discourse U. It is worth noting that these fuzzy sets come with semantics delivered by the corresponding linguistic variables. As a results, c triangular fuzzy sets with some well-defined semantics whose membership functions are characterized by $\frac{1}{2}$ overlap expressed in (4) are finally formed.

The function of encoding module is to encode each interval-valued data including in ITS $\boldsymbol{X}$ into an embedding vector with semantics so that ITS is described in terms of those linguistic variables produced already in the linguistic variable generation module. To realize this function, we can invoke directly the possibility measure presented in 2.1 to encode individual interval-valued data including in ITS $\boldsymbol{X}$. In detail, for the interval-valued data X_t emerged at time t, we first generate n probe points with the span d within the universe of discourse U. These n points together two end points of U form the ordered collection $Prob = \{prob_0, prob_1, prob_2, \cdots, prob_n, prob_{n+1}\}$, where $prob_0 = u_0$, $prob_{n+1} = u_1$ and $prob_0 < prob_1 < prob_2 < \cdots < prob_n < prob_{n+1}$ with $prob_{j+1} - prob_j = d$ $(j = 0, 1, \cdots, n)$. Next with the aid of this probe point collection, the possibility degree between the interval-valued data X_t and linguistic variables expressed in triangular fuzzy sets with membership functions characterized by $\frac{1}{2}$ overlap is calculated by the formula (1), that is,

$$\lambda_{ti} = Poss(X_t, A_i) = \max_{x \in Prob} \{\min(X_t(x), A_i(x))\}. \tag{5}$$

Thus the interval-valued data X_t is transform into an embedding vector with semantics $[\lambda_{t1}(A_1), \lambda_{t2}(A_2), \cdots, \lambda_{tc}(A_c)]$. The resulting embedding vector with semantics can be interpreted as "X_t *is* A_1 *with* λ_{t1} *possibility, or is* A_2 *with* λ_{t2} *possibility,* $\cdots$*, or is* A_c *with* λ_{tc} *possibility*". In the same manner, for other interval-valued data including in the ITS X, the corresponding possible degrees of them with respect to individual triangular fuzzy sets (linguistic variables) can be also obtained. As a result, a k-by-c possibility degree matrix associated with semantics, which is regarded as the linguistic expression of ITS X, is formed finally, viz.,

$$\boldsymbol{G} = [\lambda_{ti}(A_i)]_{\substack{t=1,2,\cdots,k \\ i=1,2,\cdots,c}} = \begin{bmatrix} \lambda_{11}(A_1) & \lambda_{12}(A_2) & \cdots & \lambda_{1c}(A_c) \\ \lambda_{21}(A_1) & \lambda_{22}(A_2) & \cdots & \lambda_{2c}(A_c) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{k1}(A_1) & \lambda_{k2}(A_2) & \cdots & \lambda_{kc}(A_c) \end{bmatrix}. \tag{6}$$

It is worth noting that the tth $(t = 1, 2, \cdots, k)$ row of the matrix $\boldsymbol{G}$ expresses that the possibility degrees that an observation X_t of ITS at time t is characterized by all linguistic variables. Whereas the ith $(i = 1, 2, \cdots, c)$ column of the matrix $\boldsymbol{G}$ expresses that the possibility degrees to which ITS X can be characterized by the ith linguistic variable A_i.

The function of inference module is to capture the relationships between the possibility degree of the corresponding interval-valued data versus individual linguistic variables at the current time $t-1$ and those degrees encountered at the next time t around the possibility degree matrix $\boldsymbol{G}$ produced by the encoding module. The relationships to be captured can be formally represented in the mapping

$$\langle \lambda_{(t-1)1}(A_1), \lambda_{(t-1)2}(A_2), \cdots, \lambda_{(t-1)c}(A_c) \rangle \to \langle \lambda_{t1}(A_1), \lambda_{t2}(A_2), \cdots, \lambda_{tc}(A_c) \rangle, \tag{7}$$

where $t = 2, 3, \cdots, k$. To realize this mapping, the artificial neural network (ANN) become an available option. Here an artificial neural network with a single hidden layer shown in Fig.4 can be considered, where the number of its input and output layers are c, which is equal to the total number of linguistic variables produced in the linguistic variable generation module. The number of neurons of corresponding hidden layer is determined experimentally by trial-and-error method.

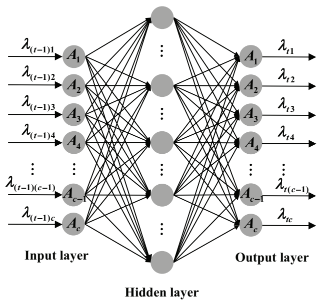


Figure 4: Architecture of artificial neural networks located in inference module.

Having the architecture of ANN, we subsequently construct the training set by organizing the possibility degree matrix $\boldsymbol{G}$ into a series of input-output pairs in the form

$$\Big\{ \Big[\overbrace{\underbrace{\lambda_{(t-1)1}, \lambda_{(t-1)2}, \cdots, \lambda_{(t-1)c}}_{\text{inputs}}}^{\text{the } (t-1)\text{th row of } \boldsymbol{G}} \Big]; \Big[\overbrace{\underbrace{\lambda_{t1}, \lambda_{t2}, \cdots, \lambda_{tc}}_{\text{outputs}}}^{\text{the } t\text{th row of } \boldsymbol{G}} \Big] \Big\}$$

with $t = 2, 3, \cdots, k$. The resulting training set is used to train the ANN in a supervised mode. With the completion of the training of ANN, the weights of connections between neurons in individual layers are obtained, which results in the realization of mapping expressed in the form of (7). When the well-trained ANN is working, it receives the embedding vector with semantics $[\lambda_{t1}(A_1), \lambda_{t2}(A_2), \cdots, \lambda_{tc}(A_c)]$ produced by the encoding module at time t, and then processes this vector in the hidden layer. The processed result is further passed to the output layer of this ANN, which results in the formation of predictive embedding vector with semantics fo time t, say $[\hat{\lambda}_{(t+1)1}(A_1), \hat{\lambda}_{(t+1)2}(A_2), \cdots, \hat{\lambda}_{(t+1)c}(A_c)]$. The predictive embedding vector with semantics reveals the estimation of possibility degree of interval-valued data X_{t+1} presented at the next time $t+1$ versus individual linguistic variables generated already in linguistic variable generation module.

The function of decoding module is to decode for the predictive embedding vector with semantics $[\hat{\lambda}_{(t+1)1}(A_1), \hat{\lambda}_{(t+1)2}(A_2), \cdots, \hat{\lambda}_{(t+1)c}(A_c)]$ produced by the inference module. Here two decoding schemes are considered:

1) The maximal possibility degree-based decoding scheme: this decoding scheme makes the model produce linguistic output. More specifically, in light of the predictive possibility degree produced by inference module, we select the linguistic variable A_j having the maximal possibility degree as the output of the model at time $t+1$, where $A_j \in \boldsymbol{A}$ with $j = \arg\max_{i=1,2,\cdots,c} \hat{\lambda}_{(t+1)i}$. Meanwhile, the output of model is interpreted as "the interval-valued data is A_j with $\hat{\lambda}_{(t+1)j}$ possibility degree at time $t+1$".

2) The possibility measure-based interval reconstruction decoding scheme: by using this decoding scheme, the model produce the output of interval format. In this decoding scheme, by means of

the linguistic variables in the linguistic variable generation module and the probe collection $Prob$ used in the encoding module, the output produced by the model, say $\hat{X}_{t+1} = [\hat{x}^L_{t+1}, \hat{x}^U_{t+1}]$, can be determined through invoking the possibility measure-based interval encoding-decoding mechanism elaborated already in Subsection 2.2 to decode the predictive embedding vector. Here $\hat{x}^L_{t+1}$ and $\hat{x}^U_{t+1}$ express the predictive lower and upper bounds of interval-valued data at time $t + 1$, respectively.

3.2 An illustrative example

In this subsection, a randomly generated ITS consisting of 12 instances, $\boldsymbol{X} = \{[0.09, 0.15], [0.21, 0.38], [0.25, 0.37], [0.32, 0.54], [0.47, 0.66], [0.42, 0.57], [0.51, 0.89], [0.48, 0.74], [0.88, 0.98], [0.73, 0.97], [0.43, 0.84], [0.31, 0.48]\}$, is used to demonstrate how to invoke the proposed method to create the corresponding ITS model. The specific modeling method is as follows:

1. *Determination of universe of discourse U*. For the ITS $\boldsymbol{X}$ to be modeled, its minimal lower bound is 0.09, and its maximal upper bound is 0.98, viz., $x_{min} = 0.09$ and $x_{max} = 0.98$. Let the trim factors l_1 and l_2 be 0.09 and 0.02, respectively. Thus the universe of discourse U can be determined, i.e., $U = [u_1, u_2] = [x_{min} - l_1, x_{max} + l_2] = [0.0, 1.0]$

2. *Formation of linguistic variables*. Suppose that the total number of linguistic variables to be generated is 5, that is, $c = 5$. Invoking the method detailed in the linguistic variable generation module, five linguistic variables can be finally formed, that is A_1 : *amplitude is lower*, A_2 : *amplitude is low*, A_3 : *amplitude is medium*, A_4 : *amplitude is high*, A_5 : *amplitude is higher*. The individual linguistic variables are described in the form of the following triangular membership functions with $\frac{1}{2}$ overlap (see Fig.4):

$$A_1(x) = \begin{cases} -4x+1, & x \in [0, 0.25) \\ 0, & otherwise \end{cases}, A_2(x) = \begin{cases} 4x, & x \in [0, 0.25) \\ -4x+2, & x \in [0.25, 0.5] \\ 0, & otherwise \end{cases}, A_3(x) = \begin{cases} 4x-1, & x \in [0.25, 0.5) \\ -4x+3, & x \in [0.5, 0.75] \\ 0, & otherwise \end{cases},$$

$$A_4(x) = \begin{cases} 4x-2, & x \in [0.5, 0.75) \\ -4x+1, & x \in [0.75, 1.0] \\ 0, & otherwise \end{cases}, A_5(x) = \begin{cases} 4x-3, & x \in [0.75, 1.0] \\ 0, & otherwise \end{cases}$$

3. *Encoding of ITS $\boldsymbol{X}$*. According to the method provided in the encoding module, we first generate an ordered probe point collection $Prob = \{0.0, 0.01, 0.02, 0.03, \cdots, 0.99, 1.0\}$ with the span $d = 0.01$ on the universe of discourse $U = [0.0, 1.0]$. Then for each instance in ITS $\boldsymbol{X}$, X_t, $t = 1, 2, \cdots, 15$, the possibility measures of them with regard to individual linguistic variable are calculated by (5) with the aid of the probe point collection $Prob$ and the linguistic variables A_i with $i = 1, 2, \cdots, 5$, which results in the formation of possibility degree matrix $\boldsymbol{G}$. The corresponding results are reported in Tab.(2), where each instance in ITS $\boldsymbol{X}$ can be interpreted in the well-defined linguistic variables. For example, the interval-valued data $[0.21, 0.38]$ presented by ITS $\boldsymbol{X}$ at time 2 can be interpreted as "*[0.21,0.38] is A_1 with 0.16 possibility or is A_2 with 1.0 possibility or is A_3 with 0.52 possibility.*"

4. *Perform inference*. According to the above-mentioned in inference module, the ANN with a single hidden layer exhibited in the form of Fig.4 is first established where the value of c is set to 5

Table 2: The matching level matrix $\boldsymbol{G}$ obtained by using five linguistic terms for ITS $\boldsymbol{X}$

Time stamp	Instance	A_1 (lower)	A_2 (low)	A_3 (medium)	A_4 (is high)	A_5 (higher)
1	$[0.09, 0.15]$	0.64	0.60	0.0	0.0	0.0
2	$[0.21, 0.38]$	0.16	1.0	0.52	0.0	0.0
3	$[0.25, 0.37]$	0.0	1.0	0.48	0.0	0.0
4	$[0.32, 0.54]$	0.0	0.72	1.0	0.16	0.0
5	$[0.47, 0.66]$	0.0	0.12	1.0	0.64	0.0
6	$[0.42, 0.57]$	0.0	0.32	1.0	0.28	0.0
7	$[0.51, 0.89]$	0.0	0.0	0.96	1.0	0.56
8	$[0.48, 0.74]$	0.0	0.08	1.0	0.96	0.0
9	$[0.88, 0.98]$	0.0	0.0	0.0	0.48	0.92
10	$[0.73, 0.97]$	0.0	0.0	0.08	1.0	0.88
11	$[0.43, 0.84]$	0.0	0.28	1.0	1.0	0.36
12	$[0.31, 0.48]$	0.0	0.76	0.92	0.0	0.0

Table 3: Training set used to train ANN

Instance No.	Input neurons (linguistic variable)					Output neurons (linguistic variable)				
	A_1	A_2	A_3	A_4	A_5	A_1	A_2	A_3	A_4	A_5
1	0.64	0.60	0.0	0.0	0.0	0.16	1.0	0.52	0.0	0.0
2	0.16	1.0	0.52	0.0	0.0	0.0	1.0	0.48	0.0	0.0
3	0.0	1.0	0.48	0.0	0.0	0.0	0.72	1.0	0.16	0.0
4	0.0	0.72	1.0	0.16	0.0	0.0	0.12	1.0	0.64	0.0
5	0.0	0.12	1.0	0.64	0.0	0.0	0.32	1.0	0.28	0.0
6	0.0	0.32	1.0	0.28	0.0	0.0	0.0	0.96	1.0	0.56
7	0.0	0.0	0.96	1.0	0.56	0.0	0.08	1.0	0.96	0.0
8	0.0	0.08	1.0	0.96	0.0	0.0	0.0	0.0	0.48	0.92
9	0.0	0.0	0.0	0.48	0.92	0.0	0.0	0.08	1.0	0.88
10	0.0	0.0	0.08	1.0	0.88	0.0	0.28	1.0	1.0	0.36
11	0.0	0.28	1.0	1.0	0.36	0.0	0.76	0.92	0.0	0.0

Table 4: Predicted output of model established by the proposed method for ITS $\boldsymbol{X}$

time stamp	Input interval	Real output	Predicted Linguistic results		Predicted interval
			linguistic variable	interpretation	
1	[0.09,0.15]	[0.21,0.38]	A_2	the amplitude of ITS is *low* at time 2	[0.25,0.38]
2	[0.21,0.38]	[0.25,0.37]	A_2	the amplitude of ITS is *low* at time 3	[0.25,0.37]
3	[0.25,0.37]	[0.32,0.54]	A_3	the amplitude of ITS is *medium* at time 4	[0.32,0.53]
4	[0.32,0.54]	[0.47,0.66]	A_3	the amplitude of ITS is *medium* at time 5	[0.47,0.63]
5	[0.47,0.66]	[0.42,0.57]	A_3	the amplitude of ITS is *medium* at time 6	[0.43,0.56]
6	[0.42,0.57]	[0.51,0.89]	A_4	the amplitude of ITS is *high* at time 7	[0.51,0.89]
7	[0.51,0.89]	[0.48,0.74]	A_3	the amplitude of ITS is *medium* at time 8	[0.49,0.74]
8	[0.48,0.74]	[0.88,0.98]	A_5	the amplitude of ITS is *higher* at time 9	[0.88,0.98]
9	[0.88,0.98]	[0.73,0.97]	A_4	the amplitude of ITS is *high* at time 10	[0.74,0.96]
10	[0.73,0.97]	[0.43,0.84]	A_4	the amplitude of ITS is *high* at time 11	[0.43,0.83]
11	[0.43,0.84]	[0.31,0.48]	A_3	the amplitude of ITS is *medium* at time 12	[0.31,0.50]

and the number of neurons in the hidden layer is set as 10 (for illustrative purpose). Next the ANN is trained in light of Tab.3, where the activation function of neurons in the input layer is a linear function. The activation function of neurons in the hidden and output layers are a sigmoid function. Once the training of ANN has been completed, the predictive embedding vector with semantics is obtained.

5. *Perform decoding.* For example, when an interval-valued data $[0.09, 0.15]$ observed at time 1 is fed into the model, with the aid of the already formed linguistic variables and probe point collection *Prob*, an embedding vector with semantics $[0.64(A_1), 0.60(A_2), 0.0(A_3), 0.0(A_4), 0.0(A_5)]$ is first obtained by encoding module and then delivered into the inference module along with the corresponding semantic neurons. With the completion of calculation, the inference module generate the corresponding output, say $[0.0(A_1), 1.0(A_2), 0.52(A_3), 0.0(A_4), 0.0(A_5)]$, which indicates the estimated possibility degree of the interval-valued data encountered at time 2 versus individual well-defined linguistic variables. Subsequently, the output of the inference module is decoded along with two different schemes: in light of the decoding scheme (i), the linguistic outcome produced by the ITS model is A_2, which means that "the amplitude of ITS is low at time 2". Whereas in light of the decoding scheme (ii), the numeric outcome produced by the ITS model is $[0.25, 0.38]$. The other predicted results are also reported in Tab.4.

4 Experimental studies

In this section, experiments concerning four real-world financial datasets with different dynamic characteristics are perform to validate the proposed modeling method, and explore the impact of the crucial parameter in the proposed method (say, the total number of linguistic variables c) on the corresponding model performance. In addition, the comparison experiments with other modeling methods of ITS is also performed to exhibit the merit of the proposed method. Before starting all experiments, the universe of discourse of each dataset is first determined. Then each dataset involved in experiments are split into the training set and the testing set in term of the proportion of 2 to 1, where the training set is used to model and the testing set is used to evaluate the performance of corresponding model.

In order to evaluate the performance of model at the linguistic level, we can envision the resulting ITS model as a classification model, viz., when an interval-valued data observed at time t is fed into the this ITS model, the interval-valued data predicted by the model at time $t+1$ belongs to which linguistic variables generated already through linguistic variable generation module. Here the subscript of the symbol expressed each linguistic variable is regarded as the corresponding class label. Thus some performance metrics used to evaluate classifier can be considered to quantify the performance of ITS model established by the proposed method on the linguistic level. In the experiments, we select the classification accuracy (L_{Accu}) and the *Kappa coefficient* (L_κ) [2] as the corresponding indicators to evaluate the performance of established ITS models at the linguistic level. They are defined as follows

$$L_{Accu} = \sum_{t=1}^{m} \frac{f(t)}{m},\ f(t) = \begin{cases} 1, & if\ \hat{L}_t = L_t \\ 0, & if\ \hat{L}_t \neq L_t \end{cases}, \tag{8}$$

and

$$L_\kappa = \frac{p_0 - p_e}{1 - p_e}, p_0 = \frac{N_{TP}}{m} \text{ and } p_e = \frac{1}{m^2}\sum_{i=1}^{c} N_{Ti} \times N_{Pi}. \tag{9}$$

In (8), m is the total number of instances, $\hat{L}_t$ and L_t are the predicted linguistic lables and true linguistic lables at time t, respectively. Here the true linguistic lables of an instance (interval-valued data) can be obtained through selecting the one having the maximal value of possibility degree in linguistic variables by means of (5). The symbols m, N_{TP}, N_{Ti} and N_{Pi} in (9) express the total number of instances, the amount of instances being correctly predicted (classified), the amount of instances truly belonging to the ith linguistic variable (class) and the amount of instances predictively belonging to the ith linguistic variable (class), respectively.

For the evaluation of performance of model at the numeric level, the mean absolute percentage error (MAPE) encountered frequently in many existing ITS models is exploited, which is defined as follows

$$MAPE = \frac{1}{m}\sum_{t=1}^{m}\left[\frac{|\hat{x}_t^L - x_t^L|}{x_t^L} + \frac{|\hat{x}_t^U - x_t^U|}{x_t^U}\right], \tag{10}$$

where m is the number of instances. $\hat{x}_t^L$ and $\hat{x}_t^U$ are the predicted lower upper bounds of interval at time t. x_t^L and x_t^U are the real lower upper bounds of interval presented at time t. Evidently, the higher the values of L_{Accu} and L_κ, the better the performance of the corresponding ITS model at the linguistic level. The smaller the values of $MAPE$, the better the performance of the corresponding ITS model at the numeric level.

Besides, when the proposed method is used to model ITS, the prob span d used in the encoding and encoding modules is set as 0.01. For the ANN located in inference module, the activation function of neurons in its input layer is linear function, and of those neurons in the hidden and output layers are sigmoid function. The number of neurons in the hidden layer is experimentally determined in the way where the number can be sought when it is enough to guarantee ANN reach minimal training error and going beyond it does not lead to a remarkable improvement of the error. Back-propagating errors method [13] is used to train ANN. To ensure high confidence in the experimental results, in all experiments, for each value of c, experiments are implemented ten times independently, and the average values of corresponding performance indicators are recorded and reported.

4.1 Summary of experimental datasets

There are four real-world financial datasets involved in the experiments, which exhibits different dynamic characteristics. The first dataset is the stock prices of General Electric Company dataset (GE). It includes 1507 observations, which records the daily lowest and highest prices from January 3, 2012 to December 29, 2017 (`https://finance.yahoo.com/quote/GE/`). The second dataset is the West Texas Intermediate crude oil spot price dataset (WTI). It consists of daily lowest and highest oil prices from January 3, 2006 to December 31, 2015, and has a total of 2525 observations (`https://cn.`

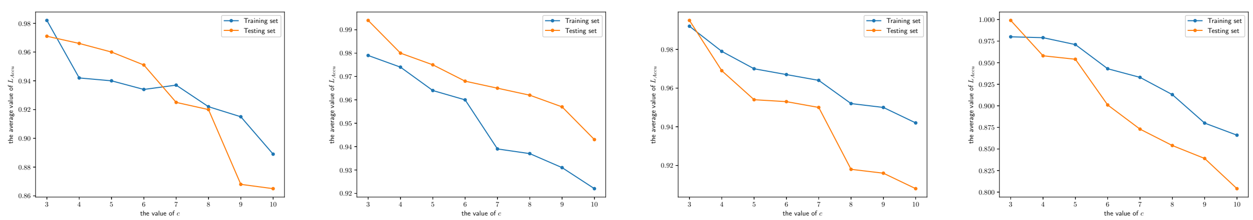
investing.com/commodities/crude-oil-historical-data). The third dataset SP500 is regard to Standard & Poor's 500 index. It records the daily lowest and highest values of Standard & Poor's 500 index from January 3, 2000 to December 31, 2013, which consists of 3521 observations (https://finance.yahoo.com/quote/%5EGSPC). The last dataset is the exchange rate of the Euro against the US dollar dataset (EURUSD) including 2322 observations. The dataset recorded daily lowest and highest exchange rates from January 1, 2007 to December 31, 2015 (https://finance.yahoo.com/quote/EURUSD=X?p=EURUSD=X). The university of discourse of above-mentioned four datasets, in turn, are $[17.0, 33.0]$, $[32, 148]$, $[660, 1850]$ and $[1.0, 2.0]$,

4.2 Experimental results and analysis

In this section, two experiments are implemented for the above-mentioned each dataset. The first experiment is to explore the impact of different value of c on the performance of ensuing ITS model, and the second experiment is to complete the performance comparisons between the ITS model established by the proposed method and the ones constructed by the other competitive methods. The corresponding experimental results are reported in Fig.5, Fig.6, Fig.7 and Tab.5, respectively.

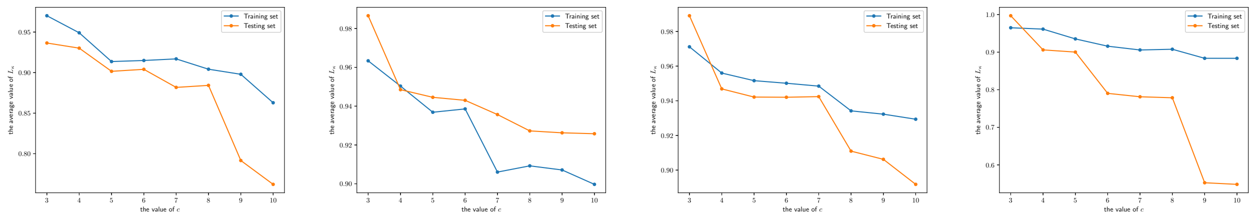
4.2.1 The impact of parameter on ITS model performance

Fig.5 clearly reveals that the average linguistic performance of established ITS model exhibits an evident downtrend with increase of value of c in terms of indicator L_{Accu}. For example, let us consider GE dataset. For its training set (see the blue dot dash line in Fig.5a), when the value of c is 3, the average value of the corresponding L_{Accu} is 0.98. When the value of c is 5, the average value decreases to 0.95. When the value of c is 8, the average value is continuously down, and decreases to 0.92. Whereas when the value of c is 10, the average value reaches its minimum, that is 0.89. For the testing set (see the orange dot dash line in Fig.5a), the similar trend is also presented — the average value of L_{Accu} is gradually down from its high value 0.97 to its low value 0.87 when the value of c increases from 3 to 10. For the remaining three datasets, the downtrend of linguistic performance of established ITS model indicated by the average value of L_{Accu} with increase of value of c can also be found in Fig.5b, Fig.5c and Fig.5d. Further, in terms of indicator L_{κ}, the linguistic performance of ITS model established by the proposed method versus the value of c is also shown a remarkable downtrend (see Fig.6) — for the training set of GE dataset, see Fig .6a, when the value of c is 3, the average value of L_{κ} is 0.97. Whereas when the value of c is gradually increased to 5, 8 and 10, the average value is declined to 0.91,0.90 and 0.86 in turn. For the testing set, the situation is also similar: when the value of c increases from 3 to 5, 8 and 10 in turn, the average value of indicator L_{κ} starts to decline from the high value 0.93 to 0.90, 0.88 and the low value 0.76. For the other three datasets, the similar trend can also be observed in Fig.6b, Fig.6c and Fig.6d. The above phenomena can be explained from two perspectives. From the perspective of expression of knowledge, the size of value of c reflects the degree of fineness of semantics expressed by fuzzy sets used to described characteristics of ITS. The higher the value of c, the finer the semantics generated in the linguistic variable generation module. This means that the linguistic variables are difficult to be allocated when assigning them into individual interval-valued data in ITS. From the perspective of classification, the more the number to



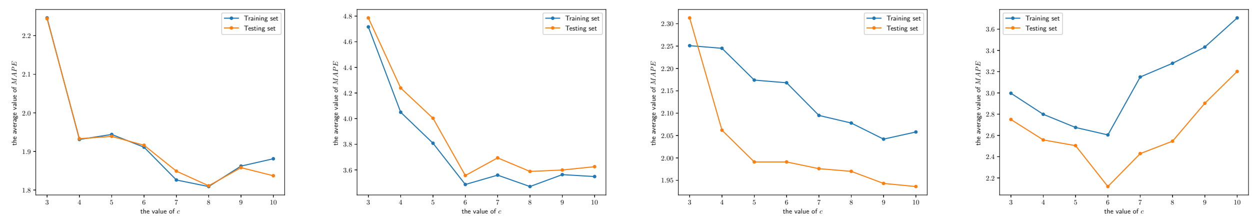
(a) GE dataset (b) WTI dataset (c) SP500 dataset (d) EURUSD dataset

Figure 5: Plots of L_{Accu} versus values of c.



(a) GE dataset (b) WTI dataset (c) SP500 dataset (d) EURUSD dataset

Figure 6: Plots of L_κ versus values of c



(a) GE dataset (b) WTI dataset (c) SP500 dataset (d) EURUSD dataset

Figure 7: Plots of $MAPE$ versus values of c

be classified, the higher the value of c. This means that the classification task becomes more difficult. It is also worth to note that the higher value of c results in the reduction of the interpretability of output of ITS model established by the proposed method.

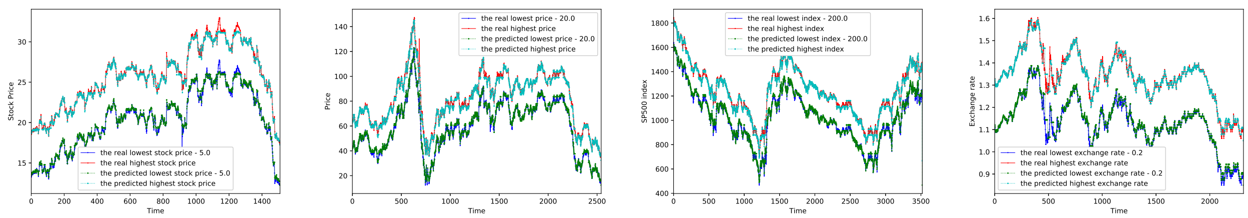
Fig.7 reveals that the relationship of the numeric performance of ITS model established by the proposed method versus different values of c — the average MAPE value does not decrease with increase of value of c. The significant decrease of the average MAPE value happens when the value of c is moved from the lower value towards some higher value. For instance, we consider the training set of GE dataset, see Fig.7a. When the value of c is 3, the average MAPE value is 2.246. When the value of c is moved to 4, the average value down to 1.931. However, when continuously moving the value of c towards its high value, the average MAPE values do not exhibit visible downtrend. At this moment, the value of c being 4 becomes a “inflection point”. For the testing set of GE dataset, the similar trend is also observed in Fig.7a. The same phenomenon is also found in Fig.7b and Fig.7c when considering WTI dataset and SP500 dataset. The corresponding inflection points are 6 and 5, respectively. We also note that for the EURUSD dataset, when the value of c moves from 3 towards 6, the average MAPE value shows a downtrend whereas the average value starts to increase when the value of c goes beyond 6. Evidently, for the EURUSD dataset, the corresponding inflection point is 6. Fig.8 shows the one-step-ahead forecasting results delivered by the corresponding ITS model established by the proposed method for each dataset, where the value of c is set as the inflection point. In order to improve visibility, in the individual subplots located in Fig.8, the lowest values are 5.0, 20.0, 200.0 and 0.2 less than the actual and predicted lowest values, respectively.

4.2.2 Comparison with other existing methods

In four real-world datasets involved in the experiments, there are three datasets, say WTI, SP500 and EURUSD datasets, allowed to directly compare with other several competitive methods in literatures [22], [10], [9], [21], [13] and [10]. Tab.5 reports that the average MAPE value delivered when using the ITS model established by the proposed method with the values of c being “inflection points” to perform one-, five-, and ten-step-ahead forecasting for these three datasets, and shows the corresponding compared results. From Tab.5, we clearly see that the ITS model established by the proposed approach can reach the better predicted results in comparison with other six hybrid models. For example, for WTI dataset, when performing one-step-ahead forecasting, the average MAPE value for the model established by the proposed method is 3.273, 6.265 for Holt^{I}-MSVR model [22], and whereas the value is 7.616 for Holt^{I}-ANN^{II} model [10], 9.589 for ARIMA^{I}-ANN^{II} model [9], 9.131 for MSVR^{I} [21], 11.385 for iMLP model [13] and 13.263 for Holt^{I} model [10]. In comparison with other models, our model can improve the average MAPE value by 47.6% for Holt^{I}-MSVR model, 57% for Holt^{I}-ANN^{II} model, 65.8% for ARIMA^{I}-ANN^{II} model, 64.1% for MSVR^{I} model, 71.3% for iMLP model and 75.3% for Holt^{I} model, respectively. When performing five-step-ahead forecasting, our model can can improve the average MAPE value by 26.7% for Holt^{I}-MSVR model, 25.3% for Holt^{I}-ANN^{II} model, 55.3% for ARIMA^{I}-ANN^{II} model, 48% for MSVR^{I} model, 59.1% for iMLP model and 64.1% for Holt^{I} model, respectively. Analogously, when performing ten-step-ahead forecasting, our model can can improve the average MAPE value by 25.7% for Holt^{I}-MSVR model, 32.5% for Holt^{I}-ANN^{II} model, 33.2% for ARIMA^{I}-ANN^{II} model, 48.6% for MSVR^{I} model, 53.8% for iMLP model and 57.5% for Holt^{I} model, respectively. For the other two datasets, the SP500 dataset and the EURUSD dataset, the same situation can also observed. Further, in comparison with other ITS models, the established model also exhibits a unique advantage, that is the model supports to form the output in linguistic format, which facilitates the understand of users to outcomes of model.

Table 5: Comparison with other methods for three datasets.

Dataset	Prediction horizon	Our approach	Holt^{I}-MSVR^{I}	Holt^{I}-ANN^{II}	ARIMA^{I}-ANN^{II}	MSVR^{I}	iMLP	Holt^{I}
WTI	1	**3.273**	6.265	7.616	9.589	9.131	11.385	13.263
	5	**5.225**	7.129	7.000	11.706	10.050	12.779	14.556
	10	**8.012**	10.785	11.872	11.994	15.602	17.345	18.859
SP500	1	**2.339**	5.641	6.721	9.674	8.336	10.631	12.306
	5	**4.614**	6.746	8.164	10.377	9.629	12.477	13.734
	10	**5.671**	8.439	11.257	11.789	14.449	15.274	17.709
EURUSD	1	**3.532**	5.793	7.304	8.903	8.502	10.515	12.260
	5	**5.379**	6.667	8.167	10.750	9.564	12.446	13.802
	10	**8.454**	10.730	12.066	11.872	15.826	17.258	19.096



(a) $c = 4$, $MAPE = 1.931$ (b) $c = 6$, $MAPE = 3.486$ (c) $c = 5$, $MAPE = 2.154$ (d) $c = 6$, $MAPE = 2.997$

Figure 8: The best forecasting results delivered by ITS models established in the case of inflection points where the value of MAPE is reported for entire dataset: (a) GE dataset, (b) WTI dataset, (c) SP500 dataset and (d) EURUSD dataset.

5 Conclusions

In this paper, a novel modeling method of ITS using possibility measure-based encoding-decoding mechanism is proposed. The method includes linguistic module, encoding module, inference module and decoding module. Here the linguistic module provides a series of linguistic variables expressed in fuzzy sets for encoding module. The encoding module can make ITS semantics in the form of possibility degree matrix which consists of embedding vectors with semantics. The inference module can realize the mapping between the embedding vectors with semantics, and the decoding module makes use of two different schemes to decode for the predictive embedding vectors with semantics produced by the inference module to form respectively the linguistic and numeric outputs. Four real-world financial ITS are involved to validate the feasibility and effectiveness of the proposed method. The corresponding experimental results draw some conclusions as follows: (i) the linguistic and numeric performance of established ITS model is highly sensitive to the number of generated linguistic variables (the value of c); (ii) the lower number of linguistic variables makes the output of the established ITS model produce better interpretability; The higher number of linguistic variables results in the degeneration of corresponding linguistic performance. (iii) in order to make the established ITS model produce better numeric accuracy, the range of number of linguistic variables is recommended as 4 to 6. The over high number of linguistic variables could result in the reduction of numeric performance of corresponding ITS model.

Funding

The research was partially supported by the National Natural Science Foundation of China under Grant Nos. 62073056 and 61876029, the National Key R&D Program of China under Grant No. 2019YFB1705103, and the Fundamental Research Funds for the Chinese Central Universities under Grant DUT20LAB129.

References

[1] Bortolan, G.; Pedrycz, W. (1997). Reconstruction problem and information granularity, *Fuzzy Systems, IEEE Transactions on*, 5(2), 234–248, 1997.

[2] Cohen, J. (1960). A coefficient of agreement for nominal scales, *Educational and psychological measurement*, 20(1), 37–46, 1960.

[3] Dubois, D.; Prade, H. (1998). Default reasoning and possibility theory, *Artificial Intelligence*, 35(2), 243–257, 1988.

[4] Dubois, D.; Prade, H. (2012). Possibility theory, In *Computational complexity*, Springer, 2240–2252, 2012.

[5] Gurney, K. (1997); *An introduction to neural networks*, CRC press.

[6] Hansen, B.E. (2000). Sample splitting and threshold estimation, *Econometrica*, 68(3), 575–603, 2000.

[7] Holt, C.C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages, *International journal of forecasting*, 20(1), 5–10, 2004.

[8] Lu W.; Zhou W.; Shan D.; Zhang L.Y.; Yang J.H.; Liu X.D. (2019). The linguistic modeling of interval-valued time series: A perspective of granular computing, *Information Sciences*, 478, 476–498, 2019.

[9] Maia, A.L.S.; Carvalho, F.D.A.; Ludermir, T.B. (2008). Forecasting models for interval-valued time series, *Neurocomputing*, 71(16-18), 3344-3352, 2008.

[10] Maia, A.L.S.; Carvalho, F.D.A. (2011). Holt's exponential smoothing and neural network models for forecasting interval-valued time series, *International Journal of Forecasting*, 27(3), 740–759, 2011.

[11] Pedrycz, W. (2008). Granular computing for data analytics: a manifesto of human-centric computing, *IEEE/CAA Journal of Automatica Sinica*, 5(6), 1025–1034, 2018.

[12] Pedrycz, W. (1994). Why triangular membership functions?, *Fuzzy Sets & Systems*, 64(1), 21–30, 1994.

[13] Rumelhart D.E.; Hinton, G.E.; Williams, R.J. (1986). Learning representations by back-propagating errors, *Nature*, 323(6088), 533–536, 1986.

[14] San Roque, A.M.; Maté, C.; Arroyo, J.; Sarabia, Á. (2007). iMLP: Applying Multi-Layer Perceptrons to Interval-Valued Data, *Neural Processing Letters*, 25(2), 157–169, 2007.

[15] Stockburger, D. (2001); *Introductory statistics: Concepts, models, and applications*, Atomic dog publishing.

[16] Sun, Y.Y.; Han, A.; Hong, Y.M.; Wang, S.Y. (2018). Threshold autoregressive models for interval-valued time series data, *Journal of Econometrics*, 206(2), 414–446, 2018.

[17] Suresh, S. and Sundararajan, N. and Savitha, R. (2013). Supervised Learning with Complex-valued Neural Networks, In *Supervised Learning with Complex-valued Neural Networks*, Springer, 49–71, 2013.

[18] Tuia, D.; Verrelst J.; Alonso L.; Pérez-Cruz F.; Camps-Valls, G. (2018). Multioutput support vector regression for remote sensing biophysical parameter estimation, *IEEE Geoscience and Remote Sensing Letters*, 8(4), 804–808, 2011.

[19] Wang, D.G.; Song, W.Y.; Pedrycz, W. (2018). A two stage forecasting approach for interval-valued time series, *Journal of Intelligent & Fuzzy Systems*, 35(2), 2501–2512, 2018.

[20] Xiong, T.; Bao, Y.K. and Hu, Z.Y.; Chiong, R. (2015). Forecasting interval time series using a fully complex-valued RBF neural network with DPSO and PSO algorithms, *Information Sciences*, 305, 77–92, 2015.

[21] Xiong, T.; Bao, Y.K.; and Hu, Z.Y. (2015). Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting, *Knowledge-Based Systems*, 55(55), 87–100, 2014.

[22] Xiong, T.; Li, C.G.; Bao, Y.K. (2017). Interval-valued time series forecasting using a novel hybrid Holt^{I} and MSVR model, *Economic Modelling*, 60, 11–23, 2017.

[23] Yang, X.S. (2009). Firefly algorithms for multimodal optimization, *Proceedings of International symposium on stochastic algorithms*, 169–178, 2009.

[24] Zhou W.; Shan D.; Yang J.H.; Lu W. (2019). The Modeling of Interval-Valued Time Series: A Method Based on Fuzzy Set Theory and Artificial Neural Networks, *International Journal of Computational Intelligence and Applications*, 18(01), 1950002, 2019.

[25] Zadeh, L.A. (1984). Making computers think like people [fuzzy set theory], *IEEE spectrum*, 21(8), 26–32, 1984.

Cite this paper as:
Lv Z., Shan D., Hu X., Lu W (2021). The Modeling of Interval-Valued Time Series Using Possibility Measure-Based Encoding-Decoding Mechanism *International Journal of Computers Communications & Control*, 16(4), 3846, 2021.
https://doi.org/10.15837/ijccc.2021.4.3846