Visual Exploration of Time-Series Forecasts Through Structured Navigation

Xiaoyi Wang
xiaoyi.wang@di.ku.dk
University of Copenhagen
Copenhagen, Denmark

Kasper Hornbæk
kash@di.ku.dk
University of Copenhagen
Copenhagen, Denmark

ABSTRACT

Evaluating the forecasting ability of time-series involves observations of multiple charts representing different aspects of model accuracy. However, the sequence of the charts observed by users is not controlled and it is difficult for users to discover relations among charts. Therefore, we propose a method for constructing a navigation structure that shows these relations based on the syntax and semantics of the charts. An excerpt from the structure is used as a context menu that allows users to navigate through a series of charts and explore their relations in a structured way. A qualitative study is conducted to evaluate the system and the results show that our approach helps users explore the connections among charts and enhances the understanding of time-series forecasting performance.

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI)**; **Visual analytics**; **Visualization systems and tools**.

KEYWORDS

model evaluation, navigation, time series

ACM Reference Format:
Xiaoyi Wang and Kasper Hornbæk. 2020. Visual Exploration of Time-Series Forecasts Through Structured Navigation. In *International Conference on Advanced Visual Interfaces (AVI '20), September 28-October 2, 2020, Salerno, Italy*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3399715.3399906

1 INTRODUCTION

Time-series forecasting is a process of using statistical models to forecast future values of a time series based on the history of that series [10]. Many businesses (e.g., stock price, retail sales, energy consumptions) rely on and benefit from the power of forecasting. Evaluating the forecasting ability of models on future data plays an important role in model selection because more accurate forecasts can result in better business decisions.

The two main activities in the evaluation are comparing model accuracy (e.g., RMSE) and comparing predicted with actual values. Showing accuracy statistics in a table is the most popular method for accuracy comparison, used by many tools (e.g., R, TiMoVA [6]). The table arranges rows as candidate models and columns as accuracy metrics. A common approach to comparing predictions is plotting the predicted and the observed values in a 2D chart. Unfortunately, users then need to evaluate models by trying to merge tables and plots.

However, a table only lists statistics and does not support exploratory analysis, and can therefore be misleading (e.g., Anscombe's quartet [1]). The situation becomes even more complex when k-fold cross-validation is applied. Then, model accuracy stops being a single measure and becomes a averaged value calculated by k-fold test sets. The reliability and robustness of models can be markedly different even if they sharing similar averaged accuracy. Although the table can list the associated standard deviations and even individual accuracy of each fold, these statistics do not facilitate the exploration.

A straightforward solution to this problem is using multiple charts to represent these statistics, which enable the exploratory analysis. This requires the observations of several charts depicting different aspects of model accuracy (e.g., averaged accuracy, individual accuracy of folds). Also, users need to observe the charts of model predictions before selecting the optimal model. The sequence of observing these charts is usually not controlled, and users have the freedom to decide the order, depending on their knowledge and exploration thus far. However, this solution does not create a clear structure of the visual exploration process, and it is not easy for users to understand the relations among charts [12]. The sequence chosen by users may not best possible reveal the connections among the charts.

Therefore, we investigate how structured navigation helps users explore the space of model accuracy, folds, and forecasts. We propose a method whereby users can systematically navigate through a series of charts representing different aspects of model performance and make comparisons. The purpose of this method is to help users better understand the hidden relationship between accuracy, folds, and forecasts.

Our threefold contributions are:

(1) a method of constructing a graph structure for structured navigation by syntax and semantics of charts,
(2) an interactive visualization tool based on the method to explore the relations among model accuracy, folds, and forecasts, and
(3) a qualitative user study to evaluate what insights users gained from our approach.

AVI '20, September 28-October 2, 2020, Salerno, Italy

ACM ISBN 978-1-4503-7535-1/20/09…$15.00
https://doi.org/10.1145/3399715.3399906

2 BACKGROUND

2.1 Model Accuracy

Time-series forecasting models are within the framework of regression models that predict continuous values instead of categorical values. Model accuracy is about summarizing forecasting errors, which are the difference between observed values and their forecasts [19]. There are several accuracy metrics commonly used to assess the model accuracy of a forecasting model, such as mean absolute percentage error (MAPE), mean absolute scaled error (MASE), root mean squared error (RMSE), and mean absolute error (MAE). Although these metrics are supported by different mathematical algorithms, all of them serve the same purpose to quantify the general performance of a model.

2.2 Methods of Evaluating Model Accuracy

The following three methods are frequently used for calculating accuracy metrics.

2.2.1 Out-of-Sample Evaluation (OOS). The method randomly splits a dataset into a training set and a test set[1]. The training set is for estimating model parameters, building, and improving models. The test set is used for calculating the scores of accuracy metrics and assessing the model performance for future data.

2.2.2 K-fold Cross-Validation (CV). A data set is divided into k-equal-sized subsets, which are also known as k folds. One of the k folds is treated as the test set, and the rest of the folds form the training set. This procedure is repeated k times. Each time, a different fold is treated as a test set. K is usually 5 or 10. The accuracy score is an averaged value over all scores of k-fold test sets.

2.2.3 Information Criterion Statistics (ICS). ICS is a general term for a set of criteria for evaluations. The most commonly used criteria are Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), and adjusted R^2. Unlike the two methods above, these criteria have rigorous theoretical justifications and mathematical formulae [20] to calculate corresponding scores without partitioning a data set.

Both OOS and ICS are conceptually simple and easy to compute, but they have potential drawbacks [20]:

(1) OOS does not make full use of the data, and the accuracy metrics are highly dependent on how the data set is partitioned.
(2) Compared to ICS, CV offers a direct estimate of model accuracy and makes fewer assumptions about the true underlying model. Moreover, ICS is not appropriate for high-dimensional data due to the difficulty of estimation of σ^2.

CV is widely used as a standard evaluation method in machine learning and is more potent than the other two methods, but practitioners often omit CV in the evaluation of time series forecasting models due to the inherent serial correlation and potential non-stationarity of time series data [4]. Recent work show that k-fold cross-validation leads to a more robust model selection for time series forecasting compared to other evaluation methods [2–4, 19].

Several works visualized CV to explain the complexity and uncertainty of models [21, 26]. However, it seems that the relationship between model accuracy and individual folds in time-series forecasting is not well explored. This study attempts to fill this gap by structured navigation.

2.3 Evaluating Model Accuracy in Time Series Forecasting

From the perspective of k-fold CV, the evaluation can be divided into two stages. The first stage uses k training sets to iteratively evaluate models with the purpose of model refinement and feature selection. Charts (e.g., ACF plot, PACF plot, and residual analysis plots) used at this stage represent the goodness of fit (strength of fit) of models, which help users investigate how well a model fits the observations [25]. A set of candidate models is selected at the end of this stage. Then, the last stage uses the k test sets to evaluate the forecasting ability of models on unseen data and select the optimal model. Charts (e.g., accuracy metrics over models, prediction over time) used in the last stage depict the ability of a model to accurately predict unseen data instead of the goodness of fit. The focus of this study is about evaluating the forecasting ability of models on future data. Therefore, charts related to the goodness of fit are not considered in this study.

In the field of time series modeling, visual analytics systems [6, 14, 15, 22] mainly focused on the first stage, and many techniques were developed to help model developers exploratory validate and refine models. For example, TiMoVA [6] is a prototype that provides visual guidance in the task of ARIMA model selection. It helps domain experts diagnose time series models by interactive plots (i.e., autocorrelation function, partial autocorrelation function, and residuals), and iteratively refine the models.

Gotz and Sun [14] proposed a method using a modified decision flow to visualize accuracy scores of predictive models by color. The method aims to help users identify problematic samples with low accuracy and problematic features associated with incorrect predictions. Similarly, Hao et al. [15] also used new visual accuracy color indicators to validate the predicted results from time series models. Even in a broader scope in machine learning, many visual analytics tools such as [11, 23] still serve the first stage with the same purpose.

Furthermore, previous work (e.g., [6, 23]) used table to evaluate model accuracy. It lists models in rows and model metrics in columns. However, the table only maps models with associated scores, which is not enough for evaluating model accuracy. Selecting the optimal model requires a visual analysis of several accuracy-related plots to better understand the triangle relationship among model accuracy, folds, and predictions.

Therefore, this study attempts to fill the gap by investigating how structured navigation can assist users in exploring the relations between accuracy, folds, and predictions to evaluate the forecasting ability of time-series models in the last stage.

[1] Sometimes three subsets include a validation set for fine-tuning model parameters, which is out the scope of this study.

3 DESIGN

This section describes the design of our tool that facilitates navigation through a series of charts in a structured way. The goal of the tool is to help users explore the relations among the charts and enhance the understanding of model forecasting abilities. We use a scenario across this section and **Section 4 Evaluation** to explain the motivation and potential tasks.

3.1 Scenario

Alex works at a retail company. He needs to forecast the percentage changes in quarterly personal consumption expenditure in the US for the next eight quarters based on a data set[2]. He chooses the 5-fold cross-validation as the evaluation method. He follows the procedure of forward-chaining [19] to split the training and test sets by setting the time window of each test set as eight quarters. Then, he uses the five training sets to train several **A**uto**R**egressive **I**ntegrated **M**oving **A**verage (**ARIMA**) models with different combinations of hyperparameters (i.e., non-seasonal parameters: p, d, q; seasonal parameters: P, D, Q). He examines the goodness of fit of the models and excluded models that either violates model assumptions or are with low accuracy. Three models remain for the last stage. He feeds the candidate models with the five test sets, and the models generate the forecasting results including accuracy statistics and predictions.

3.2 Tasks

We set the scenario and the forecasting results as the working example and completed a review of the literature [10, 19, 20, 25] to understand how to evaluate time-series models. Then, we worked with two statisticians to conclude the following tasks for evaluating the forecasting ability of time-series models. A user needs to explore the results and compare:

(1) Accuracy metrics across models (e.g., RMSE, MAPE)
(2) Accuracy metrics across folds
(3) Distance between observed and forecasted values
(4) Consistency between accuracy metrics and predictions

3.3 Data Attributes

Based on the forecasting results in the scenario, the tasks, and charts used for evaluation in literature, we summarized five attributes. Table 1 lists such five attributes and the associated descriptions. Each attribute represents one dimension of the statistical data, and each dimension can be mapped to a visual component in a chart. To construct a 2D chart, typically a numerical attribute is mapped with the y-axis, a categorical or numerical attribute is mapped with the x-axis, and one or more categorical attributes can be encoded as hue, saturation, or shape [24].

Name		Type	Representation
Model		Categorical	Candidate models (e.g., M1, M2)
Fold		Categorical	Folds represent the k test data sets
Time		Numerical	Timestamps in a specific period
Accuracy		Numerical	Accuracy metrics (e.g., RMSE)
DV	ODV	Numerical	Observed DV in the k test sets
	PDV	Numerical	Predicted DV based on the k test sets

DV: Dependent Variable **ODV**: Observed DV **PDV**: Predicted DV

Table 1: Description of Extracted Data Attributes

3.4 Syntax of Constructing charts

We aim to construct several 2D charts by using the combinations of the five attributes, which can help users complete the tasks in **Section 3.2**. The constructed 2D charts should represent different aspects of forecasting abilities. To better depict and construct the 2D charts, we use Wickham's A Layered Grammar of Graphics [28] as grammatical rules, which allows us to concisely describe the components of charts, to gain insight into the construction of the charts, and to unfold hidden connections between seemingly different charts [5, 9, 28, 29]. To simplify the written expression of the grammar, we only use the following two layers to depict the mapping between data dimensions and visual components.

3.4.1 *Layer of aesthetics.*

This layer specifies the mappings between data attributes and visual components (aesthetics) [28]. For example, we might map folds to the x-axis, RMSE (accuracy) to the y-axis, and models to hue, which can be represented in code as

$$aes(x = Fold, y = RMSE, hue = Model)$$

3.4.2 *Layer of geometric object or geom.*

Geoms explain how an abstract component will be rendered (e.g., points, lines, polygon) [28]. Together with the layer of aesthetics, this defines a statistical chart. For example, we can describe the chart in Fig. 1.3a as

$$aes(x = Fold, y = RMSE, hue = Model) + \ geom\_line$$

Based on Wickham's grammar [28] and Munzner's guidelines of visual encodings [24], we mapped the five attributes to three categories of visual components (aesthetics) in Table 2. A 2D chart can be constructed by combining a visual component from each of the three categories. The total number of combinations is 24 (3*2*4 based on the number of attributes in each category in Table 2). We mapped the attributes from each combination to the two layers in the grammar and attempted to construct the corresponding 2D chart. However, 19 combinations cannot be formed into any forms of 2D charts due to syntax errors (e.g., aes(x = Model, y = Accuracy, hue = Model)).

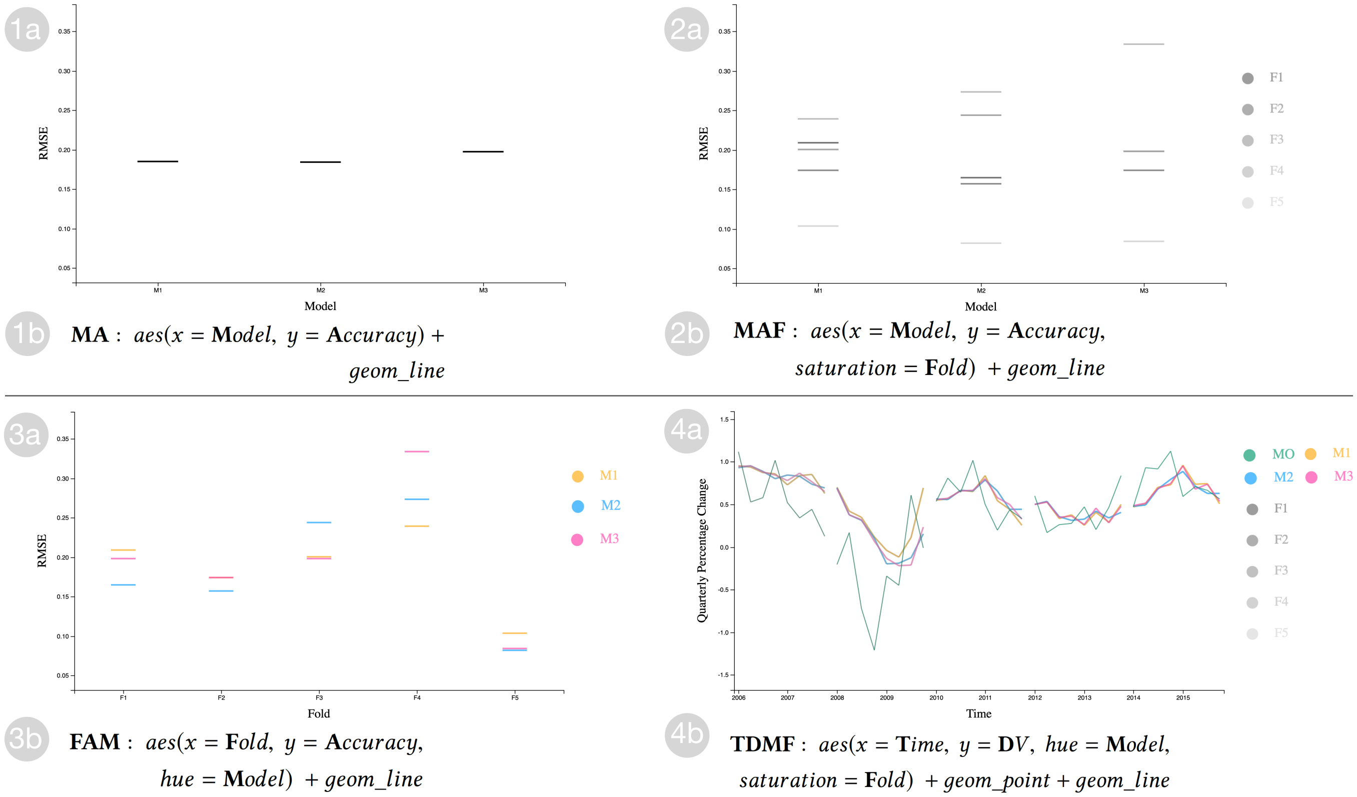
Therefore, we constructed five charts according to the rest of the five combinations and examined the semantics of each chart based on the requirements in **Section 3.2**. Two charts were found with

Visual Component	Data Attributes
X Axis as 1st Dimension	Model, Fold, or Time
Y Axis as 2nd Dimension	Accuracy, DV
Others (e.g., hues, shapes) as 3rd Dimension	Null, Model, Fold, or Model & Fold

DV: Dependent Variable

Table 2: Mapping between Visual Components and Data Attributes

[2] The data set was obtained from the fpp2 package about quarterly percentage changes in US consumption expenditure from 1996 - 2015



$\mathbf{MA}:\ aes(x = \mathbf{M}odel,\ y = \mathbf{A}ccuracy) + geom\_line$

$\mathbf{MAF}:\ aes(x = \mathbf{M}odel,\ y = \mathbf{A}ccuracy,\ saturation = \mathbf{F}old)\ + geom\_line$

$\mathbf{FAM}:\ aes(x = \mathbf{F}old,\ y = \mathbf{A}ccuracy,\ hue = \mathbf{M}odel)\ + geom\_line$

$\mathbf{TDMF}:\ aes(x = \mathbf{T}ime,\ y = \mathbf{D}\mathbf{V},\ hue = \mathbf{M}odel,\ saturation = \mathbf{F}old)\ + geom\_point + geom\_line$

Figure 1: Four Types of Charts with Grammar Notations

similar semantics, and they were considered as one chart. Finally, four types of charts were summarized from this process.

3.5 Charts Depicting the Forecasting Ability

Fig. 1 shows the four charts and the associated syntax based on the scenario in **Section 3.1**. The following part describes the grammar of each chart along with its corresponding graph shown in Fig. 1, the purpose of each chart, and the relations between charts. To effectively refer to each chart, we name each chart by combining the initial letter of each aesthetics in the grammar (e.g., **MA** represents the chart in Fig. 1.1a and its grammar in Fig. 1.1b).

MA in Fig. 1.1a and Fig. 1.1b describes the averaged forecasting accuracy of the three models.

MAF in Fig. 1.2a and Fig. 1.2b depicts the accuracy of each individual fold from the three models. Each line represents an accuracy of a fold in a model. The folds in **MAF** convey the variation of the accuracy measures compared to the averaged accuracy represented by **MAF**.

Similar to **MAF**, **FAM** in Fig. 1.3a and Fig. 1.3b also use model, accuracy, and fold to construct the chart. However, the two charts have different focuses. **FAM** helps users compare the three models by folds, while **MAF** emphasizes the comparison of folds in each model and the variation of the forecasting accuracy of a model. The two charts are transferable by switching the values of the x and color aesthetics in the grammar. Moreover, **FAM** and **MA** are also transferable by removing the third dimension $hue = model$ and change x aesthetics to $x = model$ in Fig. 1.3b.

The grammar of the three charts above depict the relationship between model accuracy, model, and fold. The accuracy in the grammar is a general term representing many different accuracy measures (e.g., RMSE, MAPE, MASE). Therefore, this grammar can yield many similar charts by replacing Accuracy with any accuracy measures for time-series forecasting. Fig. 1.1a - Fig. 1.3a are the examples of $Accuracy = RMSE$. We define these similar charts as **alternative charts**, which share the same abstract grammar.

TDMF in Fig. 1.4a and Fig. 1.4b depicts the relationship between the dependent variable (DV), time, model, and fold. A line in **TDMF** represents a fold of a model. Models and folds are encoded as hues and saturations, respectively. This chart allows users to compare the predictions of each model with the values of observed DV. The observed DV is represented as MO in the legend.

3.6 Directed Graph Structure for Navigation

Based on the syntax and semantics of the four types of charts, we propose a method for connecting these charts by a directed graph data structure that allows users to explore the hidden connections among the charts through structured navigation. Charts are linked according to the closeness of their syntax in the graph. We introduce the components of the graph in the following two subsections.

3.6.1 Vertices. Each chart is represented as a node in the graph.

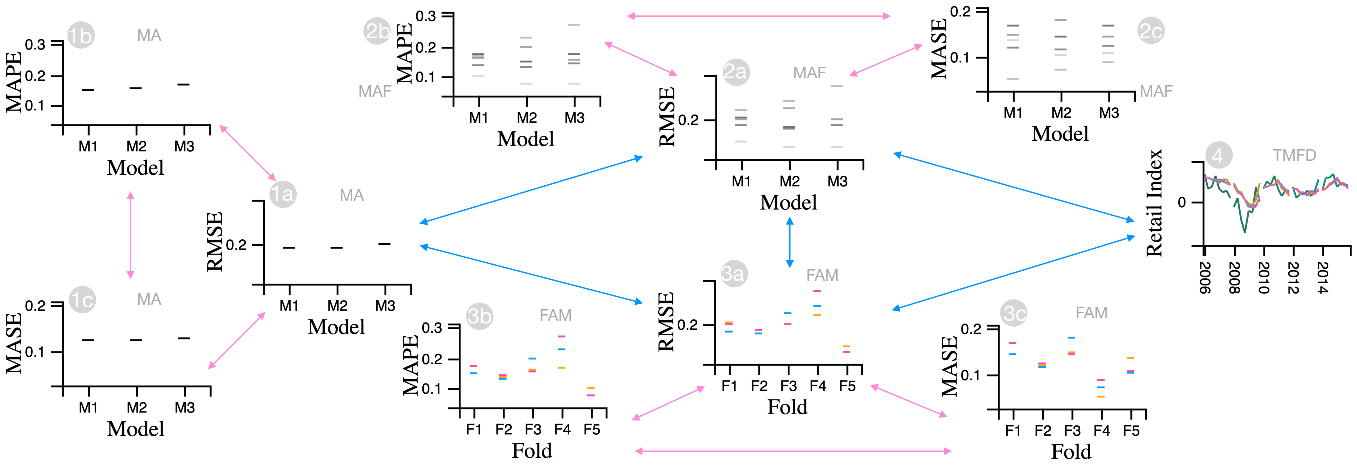


Figure 2: Graph Structure for Navigation based on the Scenario in Section 3.1

Closeness	1	2	3	4	5
NumOfChanges	5	4	3	2	1

High Closeness: 5 or 4; Medium Closeness: 3; Low Closeness: 1 or 2

Table 3: Closeness and Number of Changes

3.6.2 Edges. The closeness between two charts is measured by the number of changes of aesthetics in the syntax needed to transform one chart to the other chart. The maximum number of syntax changes in 2D space is five because five dimensions of a data set can theoretically be mapped with five visual components (i.e., x, y, hue, saturation, and shape) in a 2D chart. Therefore, we use numbers 1 to 5 to quantify the degree of the closeness and the number of changes. We also put the closeness into three categories—high, medium, and low. Charts are considered highly close when the number of changes is within two steps of changes.

Table 3 shows the relationship between the closeness of two charts and the number of syntax changes needed for the transformation. That is, the degree of closeness is inversely proportional to the number of changes. For example, Fig. 1.1a is highly close to Fig. 1.2a but not close to Fig. 1.5a because of the number of changes needed in the syntax.

The priority of connecting two charts is proportional to the level of closeness. The following steps show how to make connections among charts.

(1) **Alternative charts** with the same abstract grammar are connected. Each group of the connected **alternative charts** selects one chart as a representative, which can be connected with other charts with a different abstract grammar.
(2) A chart can be connected to another chart only if the level of closeness between the two charts is equal or higher than level of closeness between the chart and the rest of charts.

To demonstrate to the construction of the graph, we used RMSE, MAPE, and MAE as accuracy metrics, and generated three alternative charts of **MA**, **MAF**, and **FAM** respectively. Fig. 2 shows an example of connecting nine charts to form a graph structure based on the steps above. We connected the alternative charts of **MA**, **MAF**, and **FAM** and formed three groups in Fig. 2 connected by the pink arrows. Then, we selected the charts with $y = RMSE$ from the three groups as representatives awaiting for the connections with other charts in a different grammar description. In the next step, we connected the three representatives (Fig. 2.1a, Fig. 2.2a, and Fig. 2.3a) by blue arrows due to the high level of closeness. **TDMF** (Fig. 2.4) is connected with the representatives of **MAF** (Fig. 2.2a) and **FAM** (Fig. 2.3a) based on the closeness-level distance between them which is shorter than the distance between **TDMF** and **MA**.

The graph structure in Fig. 2 connects the unrelated charts according to the number of transformable changes (or level of closeness) in syntax. Users can follow a path (i.e., a directed path is a sequence of edges linking a sequence of charts in a directed graph.) to observe a sequence of changes from the starting chart to the end of chart and understand the hidden connections among the charts in the path.

3.7 Visual Design

Elmqvist et al. [12] proposed a design of structured navigation in scatterplot matrices to facilitate structured visual exploration of a multidimensional data set. The matrix serves as an overview of the dataset and allows users to explore and navigate the data set through a series of interaction techniques.

We use syntax and semantics of charts to construct a graph for structured navigation rather than the matrix. Furthermore, Elmqvist et al. [12] and other works in geographic maps [8, 18] use the overview-detail design to facilitate navigation. The overview assists users in orientation, and users can quickly switch the detailed view to a different location on the map by interacting with the overview. The purpose of our design is to facilitate the exploration of the connections among charts through structured navigation. Although the graph structure can serve as an overview for navigation, it may distract users from exploration. Therefore, we opted out of the overview and chose the context menu serving as a navigation tool.

Fig. 3 demonstrates the visual design of our navigation system. Our system consists of two main visual components—a detailed

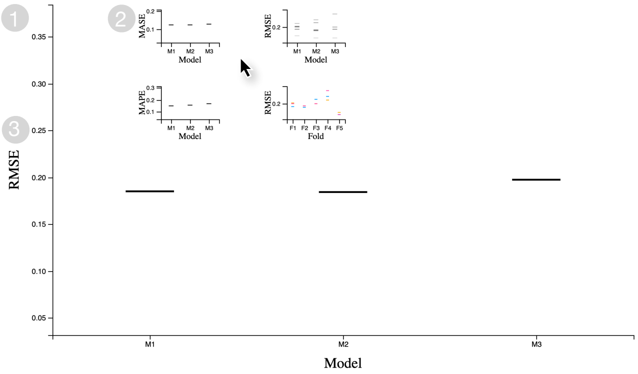


Figure 3: Overview of the System with Context Menu for Navigation

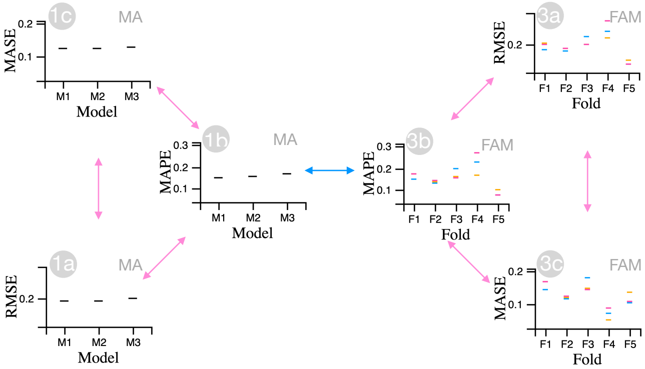


Figure 4: Dynamically Established Connection between MA and FAM

view and a context menu. The detailed view shows the currently observed chart that is one of the nine charts in Fig. 2. The context menu is based on the detailed view and shows an excerpt from the graph structure in Fig. 2. The excerpt consists of the charts that are connected with the detailed view in Fig. 2. All the charts composing the context menu are shown in the form of thumbnails instead of text descriptions. Users can immediately know the chart once they observe a thumbnail, and they do not need to map the descriptions with each chart. The context menu can be triggered by a right mouse click on the detailed view. When users left click a thumbnail view, the detailed view starts the transition to the clicked chart.

For example, Fig. 3.1 is the detailed view displaying Fig. 2.1a. Fig. 3.2 is the context menu consisting of four thumbnail views representing Fig. 2.1b, Fig. 2.1c, Fig. 2.2a, and Fig. 2.3a respectively. The four thumbnails are connected with the detailed view according to the relationships illustrated in Fig. 2.

The context menu is related to which of the nine charts is shown on the detailed view. For example, when Fig. 2.4 is shown on the detailed view, its corresponding context menu only consists of Fig. 2.2a and Fig. 2.3a.

3.8 Navigation

Users can navigate between a series of charts through the context menu. Our system supports two types of navigation in different scales.

3.8.1 Navigation within Alternative charts with the same abstract syntax. This type of navigation allows users to explore the charts only between **alternative charts** without involving other charts with different syntax in the navigation.

For example, the three charts (Fig. 2.1a, Fig. 2.1b, and Fig. 2.1c) are the **alternative charts** of **AM**. The detailed view is displaying Fig. 2.1a. Users can right click the text label for the y axis (Fig. 3.3) to show the context menu consisting of the detailed view's two **alternative charts** (Fig. 2.1b and Fig. 2.1c). Once an **alternative chart** (e.g., Fig. 2.1b, $y = MAPE$) is clicked, the detailed view starts the transition to the clicked chart Fig. 2.1b $y = MAPE$.

Meanwhile, changes also happen in the graph structure. Figure 2.1a ($y = RMSE$) loses the links to Fig. 2.2a ($y = RMSE$) and Fig. 2.3a ($y = RMSE$) . Instead, Fig. 2.1b ($y = MAPE$) replaces Fig. 2.1a ($y = RMSE$) as the representative of **MA**. Fig. 2.1b ($y = MAPE$) establishes the connections with Fig. 2.2b ($y = MAPE$) and Fig. 2.3b ($y = MAPE$). Fig. 4 displays one part of the newly established connections between Fig. 2.1b and Fig. 2.3b.

The example indicates that the graph structure is not static. This feature can reduce the steps of navigation from one alternative to another alternative chart with different syntax. The number of steps from Fig. 4.1b to Fig. 4.3b is one instead of three in Fig. 2.

3.8.2 Navigation between connected charts. Users can explore the whole structure of the graph through the context menu triggered on the detailed view. The thumbnail views on the left column of the context menu (Fig. 3.2) allows users to explore the **alternative charts** (Fig. 2.1b and Fig. 2.1c) of the detailed view. Users can explore the charts with different abstract grammar through the thumbnail views (Fig. 2.2a and Fig. 2.3a) on the right column of the context menu (Fig. 3.2).

3.8.3 Steps of Navigation. The maximum number of steps from one chart to its furthest chart is two in our design. The depth of the graph is shallow, and users can reach to any charts in the graph quickly. Therefore, this is another reason that the graph structure is opted out as an overview. We set Fig. 2.1a as the starting point of the navigation shown in the detailed view for users.

3.8.4 Animated Transitions. The animated transition can improve the understanding of the difference between statistical charts [12, 13, 16, 27]. Unlike Elmqvist et al. [12] using 3D animation, we use staged 2D animation to explain the syntax difference between two charts. We follow the guidelines of Heer and Robertson [16] to set 1 second as the animation duration for each stage. The maximum total transition time is 3 seconds since three changes are needed to transit from either FAM or MAF to TPMF.

4 EVALUATION

4.1 Rationales of the Study

The purpose of this study is to investigate how participants explore the relations between model accuracy, folds, and forecasts by structured navigation, and what insights they gain during the exploration. The evaluation for our tool is different from typical HCI studies (e.g., [18]), which usually measure accuracy in answering questions, recall of map objects, and task completion time. Quantifying either completion time or recall of objects does not help us understand how users explore and analyze forecasting accuracy statistics. Therefore, we choose think-aloud as the evaluation method in this study.

4.2 Participants

10 participants (age m = 31.3, SD = 7.5) were recruited for the study. The ratio of males to females is 7:3. Based on the self-reported expertise on evaluating time-series models, we had four novice users, four intermediate users, and two expert users. They are researchers and students with educational backgrounds in computer science and statistics from local universities. The study took approximately 40 minutes on average, and each participant received a gift (about 16.7 Euro in value based on the city-averaged wage per hour) as compensation for their participation.

4.3 Data, Forecasting Models, and Tasks

The following three data sets[3] were used in this study. D1 was only for the training purpose. D2 and D3 were used for the test session. The sequence of D2 and D3 were counter-balanced between participants.

- **D1**: quarterly visitor nights (in millions) spent by international tourists to Australia (1999-2015).
- **D2**: quarterly retail trade index in the Euro area (1996-2011).
- **D3**: quarterly percentage changes in US consumption expenditure (1996 - 2015).

We followed the procedure described in **Section 3.1** to train the models and test the models. Our tool visualized the statistics generated by the candidate models and organized them in a structured way.

Participants were asked to explore the charts by using our tool and make an analysis of the forecasting ability of the three candidate models from D2 and D3, respectively.

4.4 Procedure

The study started with a 10-minute training session. We gave a brief introduction to the study and the system. The participants were asked to try the system and were encouraged to ask questions during this session. The test session was about 20 minutes. We asked participants to evaluate the model forecasting abilities of two sets of models. Each set of models was based on one of the data sets mentioned above. Participants were asked to speak their thoughts out during this session, and we observed their operations without offering any help. We had a ten-minute interview in the last session and thanked the participants after the study. Audio and screen were recorded during the entire study.

Our system ran on Macbook Pro 2015 with a mouse and was displayed on an external 27-inch monitor with a 2560 x 1440 pixels resolution.

5 RESULTS

We used the field notes as guidance to partially transcribe the audio recordings and coded the transcription based on the typology of data models [7]. Then, we made an analysis of the transcription by affinity diagram method [17].

5.1 Exploration Through Structured Navigation

In general, the exploration path by participants can be summarized as **MA⟷ MAF⟷ FAM⟷TDMF**. They went back and forth to explore and make the analysis of the charts by structured navigation. Intermediate and expert participants tended to make a tour of all the charts through the context menu and gained an overview before a detailed analysis of individual charts. Novice participants started the analysis from one chart to another chart without the tour.

Participants found that structured navigation allows them to explore the charts in depth. For example, P2 stated, "the context menu navigated me from a simple view to a deeper view and the deepest view." P8 also found that he had the flexibility to explore different charts with increasing complexity, and the navigation helps him know what is going on with the data.

5.1.1 *Accuracy and Variation.*

Participants used **MA** to compare the averaged forecasting accuracy across the models. They switched between the three **alternative charts** (RMSE, MAPE, MASE) to check whether the performance was consistent. When it was not consistent, participants either set one metric as the primary reference (i.e., P2, P3, P5, P7, P9, P10) or visually made an average of the three measures (i.e., P1, P4, P6, P8). However, novice participants did not pay attention to the changes of y-scales when switching between three **alternative charts**. When the averaged accuracy in **MA** is similar, participants looked at the **MAF** to check the variations of each model. P8 and P10 also used **FAM** to compare the variations of three models folds, but most of the participants compared the model performance per fold by **FAM**. They used **FAM** to identify a fold as outlier where models performed the worst. P9 said, "This chart (**FAM**) helps me better observe folds and find outliers. It is easy to check the forecasting consistency of models by this chart."

Also, four participants only used the context menu for the y-axis to switch between **alternative charts**. The rest of the participants mainly used the context menu on the detailed view to switch.

5.1.2 *Prediction.*

We found that participants used **TMFD** to visually analyze the distance between observed values and predicted values, and to confirm their findings from previous charts. For example, P3 and P6 frequently switched between **TMFD** and **FAM** to compare the distance between predicted and actual values. P3 stated that the two charts are similar that he can easily map each fold in **FAM** with each time-series segment in **TMFD**. It is useful for him to analyze model performance by combining the accuracy

[3] The three sets were obtained from the fpp2 R package.

of each fold with the distance between observed and prediction values.

5.2 Understanding the Relations between Charts

We found that participants used structured navigation to understand the relations between model accuracy, folds, and predictions. For example, P1 stated, "The prior knowledge gained from previous charts I observed helps me to understand the prediction chart (**TDMF**)."

P1, P2, P4, P7, P9 found that they can evaluate the forecasting ability from different perspectives through structured navigation. P9 stated, "I can look at the accuracy by folds, by models, and by predictions." P3 said that he followed the menu, found patterns between charts, and explored different types of charts and different types of accuracy metrics.

5.3 Animated Transition

All participants found that the animated transition helps them understand the difference between the two charts. For example, the spreading-out folds made P1, P5, P6, P8 realize the variations of three models when **MA** is transiting to **MAF**.

P2 stated, "At the beginning, I did not understand the meaning of the averaged accuracy (**MA**), but when I observed the transition from **MA** to **MAF**. I understand the meaning of both charts." P4 said, "The animation helps me notice the changes of the y-scales when I switched from one accuracy metric to another metric."

5.4 Thumbnail view

Participants found the thumbnail views intuitive and self-explanatory. P5 stated, "The context menu is very intuitive, and I know where to go when I see the thumbnail views." Thumbnail views also facilitated the comparison between charts with different accuracy metrics. P4 said that "even though the views are mini, I can still observe the difference between RMSE, MAPE, and MASE. The big view (detailed view), together with the mini views (thumbnails) are similar to putting views side by side. Sometimes I don't need to switch. I can directly compare them by the mini views".

5.5 Context Menu

All participants did not feel lost without an overview map during the navigation. They felt the control of the navigation, and they can see different angles of the model accuracy by structured navigation. For example, P1 stated, "I don't feel I got lost. I can select where to go. I am in control of the navigation". P7 stated, "I'd rather click a few more times (to get the desired chart) than not know exactly where to go. If I have all the options at once, I will be confused." However, P10 preferred to see a context menu with all the charts in thumbnails at once.

6 DISCUSSION AND FUTURE WORK

We have presented a method for constructing structured navigation interfaces. The use of the method is illustrated with a tool to explore time-series models. The tool aims at supporting the exploration of the relations among charts and thereby allow users to build up an understanding of the models. By following the steps described in this paper, the method is applicable to many other data sets, both concerning time series but also other types of data. The qualitative evaluation suggests that our tool supports this kind of exploration and facilitates the evaluation of how well time-series models forecast observations. However, some points remain unexplored; we discuss three of them next.

Scalability. This study focused on structured navigation for evaluating the forecasting ability of time series. We did not investigate how structured navigation can help users validate the goodness of fit of models in the k-fold training sets. Our method should be scalable to generate the graph structure. However, the size of the graph of charts may become quite large, because more types of charts are introduced. For instance, we might run out of space in context menus due to a large number of charts being closely associated with the current chart. Clustering charts or having multi-layered menus might be solutions to this problem.

Also, as the number of models and folds increases in a 2D chart, we may run out of the visually distinguishable options for the hue and saturation encodings. We need to investigate new encoding methods to address this problem.

Interaction. We did not provide many interaction techniques in this study because we hope that participants only focus on the navigation and the understanding of relations between charts without introducing biases. In the future, we need to design proper interactions (e.g., filtering, zooming) to facilitate comparison and exploration.

Different user groups. Our tool aims to serve users with different expertise. The evaluation shows that our tool supports this purpose. However, we also observed that participants with different expertise behaved differently. Therefore, we need to investigate further how different user groups use the tool and make proper designs for them.

7 CONCLUSION

In this paper, we proposed a method of constructing a graph structure that connects multiple charts based on their syntax and semantics. Then, we developed a visualization tool using an excerpt from the graph as a context menu, which helps users navigate through a series of charts depicting different aspects of the time-series model forecasting ability. A think-aloud user study was conducted to evaluate the tool, and the study suggests that structured navigation facilitates the visual exploration of the relations among model accuracy, folds, and predictions. Thus, the tool enhances the understanding of the forecasting performance of time-series models.

ACKNOWLEDGMENTS

The authors would like to acknowledge Innovation Fund Denmark and the BIOPRO2 strategic research center (Grant No. 4105-00020B).

REFERENCES

[1] F. J. Anscombe. 1973. Graphs in Statistical Analysis. *The American Statistician* 27, 1 (1973), 17–21. https://doi.org/10.1080/00031305.1973.10478966

[2] Sylvain Arlot and Alain Celisse. 2010. A survey of cross-validation procedures for model selection. *Statist. Surv.* 4 (2010), 40–79. https://doi.org/10.1214/09-SS054

[3] Christoph Bergmeir and José M. Benítez. 2012. On the use of cross-validation for time series predictor evaluation. *Information Sciences* 191 (2012), 192 – 213. https://doi.org/10.1016/j.ins.2011.12.028 Data Mining for Software Trustworthiness.

[4] Christoph Bergmeir, Rob J. Hyndman, and Bonsoo Koo. 2018. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis* 120 (2018), 70 – 83. https://doi.org/10.1016/j.csda.2017.11.003

[5] Jacques Bertin. 2010. *Semiology of Graphics: Diagrams, Networks, Maps.* Esri Press, Redlands. ISBN: 978-1-589-48261-6.

[6] M. Bögl, W. Aigner, P. Filzmoser, T. Lammarsch, S. Miksch, and A. Rind. 2013. Visual Analytics for Model Selection in Time Series Analysis. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec 2013), 2237–2246. https://doi.org/10.1109/TVCG.2013.222

[7] In Kwon Choi, Taylor Childers, Nirmal Kumar Raveendranath, Swati Mishra, Kyle Harris, and Khairi Reda. 2019. Concept-Driven Visual Analytics: An Exploratory Study of Model- and Hypothesis-Based Reasoning with Visualizations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article 68, 14 pages. https://doi.org/10.1145/3290605.3300298

[8] Andy Cockburn, Amy Karlson, and Benjamin B. Bederson. 2009. A Review of Overview+detail, Zooming, and Focus+context Interfaces. *ACM Comput. Surv.* 41, 1, Article 2 (Jan. 2009), 31 pages. https://doi.org/10.1145/1456650.1456652

[9] D. R. Cox. 1978. Some Remarks on the Role in Statistics of Graphical Methods. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 27, 1 (1978), 4–9. https://doi.org/10.2307/2346220

[10] J. D. Cryer and K. S. Chan. 2008. *Time Series Analysis: With Applications in R.* Springer. https://books.google.dk/books?id=MrNY3s2difIC

[11] D. Dingen, M. van't Veer, P. Houthuizen, E. H. J. Mestrom, E. H. H. M. Korsten, A. R. A. Bouwman, and J. van Wijk. 2019. RegressionExplorer: Interactive Exploration of Logistic Regression Models with Subgroup Analysis. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan 2019), 246–255. https://doi.org/10.1109/TVCG.2018.2865043

[12] N. Elmqvist, P. Dragicevic, and J. Fekete. 2008. Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (Nov 2008), 1539–1148. https://doi.org/10.1109/TVCG.2008.153

[13] Gapminder. [n.d.]. Unveiling the beauty of statistics for a fact based world view. https://www.gapminder.org/

[14] David Gotz and Jimeng Sun. 2014. Visualizing Accuracy to Improve Predictive Model Performance. *IEEE VIS Workshop on Visualization for Predictive Analytics, Paris, France.*

[15] M. C. Hao, H. Janetzko, S. Mittelstaedt, W. Hill, U. Dayal, D. A. Keim, M. Marwah, and R. K. Sharma. 2011. A Visual Analytics Approach for Peak-Preserving Prediction of Large Seasonal Time Series. *Computer Graphics Forum* 30, 3 (2011), 691–700. https://doi.org/10.1111/j.1467-8659.2011.01918.x

[16] J. Heer and G. Robertson. 2007. Animated Transitions in Statistical Data Graphics. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov 2007), 1240–1247. https://doi.org/10.1109/TVCG.2007.70539

[17] Karen Holtzblatt and Hugh Beyer. 2016. *Contextual design: Design for life.* Morgan Kaufmann.

[18] Kasper Hornbæk, Benjamin B. Bederson, and Catherine Plaisant. 2002. Navigation Patterns and Usability of Zoomable User Interfaces with and Without an Overview. *ACM Trans. Comput.-Hum. Interact.* 9, 4 (Dec. 2002), 362–389. https://doi.org/10.1145/586081.586086

[19] Robin John Hyndman and George Athanasopoulos. 2018. *Forecasting: Principles and Practice* (2nd ed.). OTexts, Australia.

[20] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2014. *An Introduction to Statistical Learning: With Applications in R.* Springer Publishing Company, Incorporated.

[21] J. Krause, A. Perer, and E. Bertini. 2014. INFUSE: Interactive Feature Selection for Predictive Modeling of High Dimensional Data. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (Dec 2014), 1614–1623. https://doi.org/10.1109/TVCG.2014.2346482

[22] T. Löwe, E. Förster, G. Albuquerque, J. Kreiss, and M. Magnor. 2016. Visual Analytics for Development and Evaluation of Order Selection Criteria for Autoregressive Processes. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (Jan 2016), 151–159. https://doi.org/10.1109/TVCG.2015.2467612

[23] T. Mühlbacher and H. Piringer. 2013. A Partition-Based Framework for Building and Validating Regression Models. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec 2013), 1962–1971. https://doi.org/10.1109/TVCG.2013.125

[24] Tamara Munzner. 2014. *Visualization analysis and design.* AK Peters/CRC Press.

[25] G. Shmueli and K.C. Lichtendahl. 2016. *Practical Time Series Forecasting with R: A Hands-On Guide [2nd Edition].* Axelrod Schnall Publishers. https://books.google.dk/books?id=-xWXDwAAQBAJ

[26] C. Turkay. 2014. Visualizing Time Series Predictability. In *IEEE VIS 2014 Workshop on Visualization for Predictive Analytics, Paris, France.* https://openaccess.city.ac.uk/id/eprint/4366/

[27] F. B. Viegas, M. Wattenberg, F. van Ham, J. Kriss, and M. McKeon. 2007. ManyEyes: a Site for Visualization at Internet Scale. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov 2007), 1121–1128. https://doi.org/10.1109/TVCG.2007.70577

[28] Hadley Wickham. 2010. A Layered Grammar of Graphics. *Journal of Computational and Graphical Statistics* 19, 1 (2010), 3–28. https://doi.org/10.1198/jcgs.2009.07098

[29] Leland Wilkinson. 2005. *The Grammar of Graphics (Statistics and Computing).* Springer-Verlag, Berlin, Heidelberg.