Context Aware Recommendation for Data Visualization

W. A. D. Kanchana  
kanchana.12@cse.mrt.ac.lk  
G. D. L. Madushanka  
gdlmadushanka@gmail.com  
H. P. Maduranga  
prasad.12@cse.mrt.ac.lk  
M. D. M. Udayanga  
udayanga.12@cse.mrt.ac.lk  
D. A. Meedeniya  
dulanim@cse.mrt.ac.lk  
G. I. U. S. Perera  
indika@cse.mrt.ac.lk

Department of Computer Science and Engineering  
University of Moratuwa  
Moratuwa, Sri Lanka

ABSTRACT
Visualization plays a major role in data mining process to convey the findings properly to the users. It is important to select the most appropriate visualization method for a given data set with the right context. Often the data scientists and analysts have to work with data that come from unknown domains; the lack of domain knowledge is a prime reason for incorporating either inappropriate or not optimal visualization techniques. Domain experts can easily recommend commonly used best visualization types for a given data set in that domain. However, availability of a domain expert in every data analysis project cannot be guaranteed. This paper proposes an automated system for suggesting the most suitable visualization method for a given dataset using state of the art recommendation process. Our system is capable of identifying and matching the context of the data to a range of chart types used in mainstream data analytics. This will enable the data scientists to make visualization decisions with limited domain knowledge.

CCS Concepts
• Human-centered computing → Visualization → Visualization application domains → Information visualization
• Information systems → Information systems applications → Data mining.

Keywords
Recommender systems; Data visualization; Context awareness; Machine learning; Rule based engine

1. INTRODUCTION
Information technology plays a major role in modern daily life. There is a tendency to collect user details by many e-commerce companies such as Amazon, eBay and professional networks such as LinkedIn, ResearchGate in order to provide suggestions for users, identify different design patterns, etc. Manufacturers and sellers such as YouTube, Amazon and eBay are increasingly interest in collecting information, which is related to their customers. In interactive machine learning, users interact with the learning algorithm to solve problems.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

ICCIP ’16, November 26-29, 2016, Singapore, Singapore  
© 2016 ACM. ISBN 978-1-4503-4819-5/16/11...$15.00  
DOI: http://dx.doi.org/10.1145/3018009.3018027

Training and recommendations of such systems require expert knowledge and complex mechanisms. Thus, it is important to have interactive systems with continuous feedback on the learning process and performance of the model being used. The decision making process and the visual analysis of such systems are supported by data visualization. A visual representation facilitates better accessibility, understandability and usability of data and supports clear and efficient communication.

Although there are many visualization tools, most of them do not focus on the context the data when visualizing data [1, 2]. In this paper we introduce a framework that focuses on the context of the data and recommend the best visualization technique for the given dataset. We provide a mechanism to characterize user data and map into a group of matching visualization techniques. Context awareness (CA) component extracts the context/ domain specific metadata from a given dataset and sends to the rule based recommendation component. Each domain has its own way of depicting data; so that, the visualization is highly dependent on the targeted domain [3]. Recommendation component uses machine learning (ML) techniques to reveal hidden features when selecting a visualization technique for a data set. Rule based (RB) engine is used to refine recommendations provided by the ML component. Then we visualize the processed data using the selected visualization techniques and front end visualization libraries such as chart.js, Google chart libraries and D3.js.

The paper is structured as follows. Section 2 describes related work. Section 3 explains the features of the proposed system and Section 4 details the implementation. Proposed User Interfaces are shown in Section 5 and Section 6 concludes the paper.

2. BACKGROUND
Different tools and techniques are available for data management and visualization. RapidMiner [2] is a powerful tool than Weka [4] that used for machine learning, data mining, predictive analytics, etc. RapidMiner enables rapid prototyping and application development and allows easy implementation of ETL process (Extract, transform, load) [5]. Its workflow architecture eliminates the need for user level programming to process data. Each operator’s inputs and outputs are well defined and a process can be created by linking operators in a workflow. However, this does not provide automated data context identification.

Google Fusion Table [6] is a data management application targeted for the users without an expert knowledge. It supports features such as data acquisition, collaboration, visualization and web publishing. For example, map views for geographic data; timeline and motion charts for date and time type data; bar charts and pie charts for data with numeric columns.
VizRec [7] is a system that learns user profiles and their feedback to give recommendations. They have gathered data through crowdsourcing. They use a hybrid approach by combining a Rule based RS and collaborative filtering RS. This is fairly similar approach what we have been working with, yet it differs from our proposed system by the second part. Instead of collaborative filtering of user profiles we use content based filtering of charts.

3. SYSTEM FEATURES
Main intention of this work is to recommend the most suitable visualization technique for a given data set. User does not need to worry about their knowledge or experience on contextual visualization of analyzed data. The system will fulfill the knowledge gap between user and the data analytic expertise. Once user uploads the data to the system they can select columns for visualization. Then a user receives a set of recommendations for the given data set. Users can select best matching technique out of those and rate the recommendations. Those ratings will be stored in order to refine the recommendation process. This Graphical User Interface (GUI) is provided for normal users. Also, an API is provided for developers to interact with the system. System administrators can add new chart types by defining the characterization of particular chart type. Depending on the user feedback to chart recommendations rule adjustments are generated automatically. Administrators can decide to accept or reject those adjustments to improve the system. If a given adjustment is recommended frequently that means the change is important. After user accepting a rule adjustment those changes are added to the rule engine.

4. SYSTEM METHODOLOGY
4.1 Information Processing Flow
The functionality of a web application created on top of our framework using the given API is depicted in the data flow diagram shown in Figure 1. First the user uploads data set to the web application. Then the data set is moved to the Context Awareness (CA) component, which recognizes data types, continuous and discrete nature and some special context features such as date time, percentage values, and geo location data along with the inter column dependencies of data. Then the user is prompted with the recognized context and asked for alterations and confirmations. This user feedback is used to improve the CA component. After user’s confirmation is recognized the context object is sent to Machine Learning (ML) and Rule based (RB) components. Those components will use that to recommend matching chart types. In initial stages results from the model may be noisy and may include false positives. Therefore rule engine is there to maintain the accuracy while model become consistent gradually. The system does not use only the rule engine as a helping module, but it also gets improved by the usage through its own learning. Necessary rule adjustments are automatically generated by the system and developer can accept or reject them to improve rules. Finally, a recommending list of charts in the order of importance is given to the user. User can apply them, see the visualization and finally select best match for the data set. Feedback is also recorded to be used for the subsequent iterations of training the model. Further, developers can submit new charts types and new rule definitions to the system. They can receive auto generated rule adjustments and accept or reject them to improve the system. These tasks may not occur frequently and not mandatory to the functionality of the system.

4.2 Graph Characterization
In order to provide chart recommendations we have encoded charts and user input data sets using common mechanism. Chart or data set is characterized using their number of dimensions, intension (comparison, composition, relationship, distribution) of the visualization and the properties of each dimension [8]. Those dimensional properties include cardinality, continuous/ discrete nature, context of the data (numerical, nominal or ordinal) and the partial orderings exist between columns [9].
4.3 Major System Components

4.3.1 Context Aware Component
This CA component is consists of Schema Extractor and Entity Extractor components. The Schema Extractor identifies the schema of a given data table in the format of CSV, MYSQL, or XML etc. This component is capable of recognizing all input data formats. For some of the databases, schema is already given with the dataset (such as sql file). The dataset schema can be easily identified by analyzing metadata of the given dataset. Schema extractor module goes through the entire record set and checks it with possible matching string, which is given in a lookup table. Lookup table contains possible string and each string mapped to relevant ontology. Levenshtein algorithm [10] is used to handle string misspelling and identify relevant string from given lookup table. Based on the user feedback, lookup table gets updated with new string and mapped ontology. Also, new entities are added to the lookup table for better extraction.

The Entity Extractor identifies the type of entity given in a column. This module goes through the selected cells in each column and separates it into different ontologies. First, Random sampling is applied on data as a sampling technique to reduce execution time with the process. Then each data cell is loaded into the regular expression filter. This filter puts strings into regular expression entity recognition and extract entities such as date, time, and currency etc. Remaining unrecognized strings are passed into DBpedia Simple Protocol and RDF Query Language (SPARQL) Named Entity Recognition (NER) module. This module creates SPARQL statement and sends it to DBpedia server by using HTTP request. DBpedia dictionary looks up for the string and returns possible ontology. This interface can return query result as HTML, XML, JSON, CSV etc. Here, we have used JSON as it is lightweight. SPARQL module analyzes the returned JSON object and identifies related ontology. This SPARQL interface can be also used to get related information for the dataset such as geo coordinate for given location.

4.3.2 Recommendation Component
Once context of each column is identified with the Entity Extractor component, this information is fed to recommendation engine for relevant visualization suggestions. Recommendation is handled by mainly two components: Model developed using Machine Learning and Rule based recommendation engine. In some cases, a given set of rules may not define a clear separation between chart types, which depend on the cardinality of dimensions. Here, a ML model which is trained using practical instances of visualization recommendations is used, as it is hard to define an exact threshold for switching between chart types. However, in some cases it may recommend irrelevant and incorrect results. In such situations the RB component, becomes dominant to produce the recommendation list. This RB component can also be considered as a filter for results produced by the previous model. With that, system becomes more robust under anomalies and faulty user feedbacks (Since system uses user feedback to improve the model, incorrect user feedback may lead to incorrect recommendations).

This ensures the robustness of the framework where system will not provide extremely irrelevant recommendations under any circumstances. When considering the sustainability of the framework, it should be easy to add new chart types. With the usage and usefulness, new types will be appeared at the top in the recommendations.

4.3.2.1 Machine Learning Component
The Machine Learning (ML) component reveals hidden features when selects a visualization technique for a data set. This component makes it easy to add new chart types to the framework and helps users to get relevant chart types for their data set. First, the ML component reads data from a data set, which consists of mapping between different contexts and corresponding visualizations. Then it fits those data to a model. A probability based decision tree approach is implemented for the classification of the most relevant chart type for particular data set. Performance of this component is highly dependent on the output of the CA component. Since the internal process information is dependent on the metadata collected on the columns, the data set is also a critical aspect of this component. Maturity of the data set is important as the coverage of model directly depends on the data set. User selections and ratings are used to improve the data set in turn improves the whole system of recommendations.

4.3.2.2 Rule Based Component
The RB component refines recommendations provided by the ML component. In a cold start situation ML component may provide incorrect results. In such case, RB component provides better results based on specified set of rules to filter irrelevant results. This type of failure is possible since ML component depends on user feedback and users may provide misleading feedbacks intentionally or mistakenly. First, the RB component reads context of the data set given by the user. Then with the defined rules it funnels the possible chart types for visualization. Next it considers the list provided by the ML component and its confidence of recommendation. Using the list generated by rule based engine, it filters and refines the list generated by the ML component. If the confidence of this result is low, then the new list will dominate the final recommendation list.

4.4 Implementation View
Figure 2 shows the layered view of the web application. Presentation layer consists of a set of User Interfaces (UI) to interact users with the system easily; Data upload UI to upload files, Column selection UI to select set of columns to be visualized and to apply changes to the automatically recognized context, recommendation and chart view UI to show the ranked order of chart recommendation and apply those chart types to
data, feedback UI to collect user feedbacks. For developers there is a UI to add new charts and accept and reject rule adjustments. Logical layer provides the core functionality including context aware rule engine, model trainer, rule adjustment generator, and visualization prediction components. Moreover, persistent data and the business logic view components are separated by a data access layer, which provides access to user feedback, chart context, rule definition databases and finally to the train models.

5. USER EXPERIENCE
The proposed system consists of set of Graphical User Interfaces (GUIs) that facilitate user interaction.

Data uploading interface of this system is shown in Figure 3. When a dataset is uploaded, the system will preview the content of the dataset as a table. The user can select the columns of this table (i.e., each column represents an attribute of the dataset) which need to be visualized. The Recommendation GUI with the generated recommended chart types is shown in Figure 4. Users can preview a chart, which is identified as the appropriate recommendation for a selected dataset by hovering the mouse over each recommendation. Also, the preview shows the ratings of the previous users (how many users have used this particular chart type to visualize similar kind of data sets previously).

![Figure 3: Data Uploading Window](image)

![Figure 4: Recommendation Chart Types Window](image)
The user feedback window shown in Figure 5 visualizes the data set with the selected chart type by the user. After applying recommended visualizations user select the best visualization method applicable for their data. Then the user can rate that visualization as a measure of to which extent the chart type represents the required information and to which extent it correctly represents the context of the dataset. Here, a user can download the chart in a preferred content media format from a range of media format types such as PNG, JPEG, etc.

6. CONCLUSION
As the interest on data science is increasing rapidly there is a need for tools which are user friendly, accurate and powerful enough to support for data visualization fulfilling a broad spectrum of user needs. This paper introduced a new paradigm of the data visualization using the machine learning based context identification. The framework and the API provided through implementation are capable of recommending the most appropriate chart types to a given dataset by considering the context of each dimension of the data. This framework will become a prominent tool between the data analyzers and the open source community.

While this research is work in progress, we aim at providing a complete recommender platform which is generic to data visualization need for a given data analytic project through the subsequent releases. As of present state of the art in the field a solution of this nature with domain and context aware recommendations for best fitting data visualization for universal use-cases is not yet available beyond conceptual suggestions. Therefore, it is believed that the proposed framework in this research will bring in a uniquely positive contribution to help visualize complex datasets and data analytic results irrespective of whether users have expertise to do so manually. As a result the work presented in this paper will useful to the fields of user experience (UX) and data analytics at large scale.

7. REFERENCES