Visualization and Pattern Identification in Large Scale Time Series Data

Steve Holtz
Computer Science
Kean University

Guillermo Valle
Computer Science
Kean University

Jessica Howard
NJCSTM
Kean University

Patricia Morreale
Computer Science
Kean University

ABSTRACT

Visualization of massively large datasets presents two significant problems. First, the dataset must be prepared for visualization, and traditional dataset manipulation methods fail due to lack of temporary storage or memory. The second problem is the presentation of the data in the visual media, particularly real-time visualization of streaming time series data. An ongoing research project addresses both these problems, using data from two national repositories. This work is presented here, with the results of the current effort summarized and future plans, including 3D visualization, outlined.

KEYWORDS: Visualization, data streams, time series data.

INDEX TERMS: H.2.8 [Database Management]: Database Applications—Data Mining, Spatial databases and GIS; H.5.1 [Information Interfaces]: Multimedia information systems-virtual reality.

1 INTRODUCTION

Visualization of data patterns, particularly 3D visualization, represents one of the most significant emerging areas of research. Particularly for geographic and environmental systems, knowledge discovery and 3D visualization is a highly active area of inquiry. Recent advances in association rule mining for time series data or data streams makes 3D visualization and pattern identification on time series data possible.

In streaming time series data the problem is made more challenging due to the dynamic nature of the data. Emerging algorithms permit the identification of time-series motifs [1] which can be used as part of a real-time data mining visualization application. Geographic and environmental systems frequently use sensor networks or other unmanned reporting stations to gather large volumes of data which are archived in very large databases [2]. Not all the data gathered is important or significant. However the sheer volume of data often clouds and obscures critical data which causes it to be ignored or missed.

The research presented here outlines an ongoing research project working to visualize the data from national repositories in two very large datasets. Problems encountered include dataset navigation, including storage and searching, data preparation for visualization, and presentation.

2 THEORY

Data filtering and analysis are critical tasks in the process of identifying and visualizing the knowledge contained in large datasets, which is needed for informed decision-making. This research is developing approaches for time series data which will permit pattern identification and 3D visualization. Research outcomes include assessment of data mining techniques for streaming time series data, as well as interpretive algorithms, and visualization methods which will permit relevant information to be extracted and understood quickly and appropriately. Kean University's CAVE will be used to provide an immersive virtual reality environment for the visualization.

2.1 Large scale data for visualization

This research works with datasets from the National Oceanic and Atmospheric Administration **(**NOAA**),** a federal agency in the U.S., focused on the condition of the oceans and the atmosphere. The purpose of the research project is to take meteorological data and analyze it to identify patterns that could help predict future weather events. Data from the GHCN (Global Historical Climatology Network) dataset [3] was initially used. Earlier research had worked with NOAA's Integrated Surface Dataset (ISD) [4]. Both of these datasets are open access and the volume of streaming time series data was significant and growing.

The GHCN dataset consists of meteorological data from over 76,000 stations worldwide with over 50 different searchable element types. Examples of element types include minimum and maximum temperature, precipitation amounts, cloudiness levels, and 24-hour wind movement. Each station collects data on different element types.

2.2 Time series data analysis

Searching for temporal association rules in time series databases is important for discovering relationships between various attributes contained in the database and time. Association rules mining provides information in an "if-then" format. Because time series data is being analyzed for this research, time lags are included to find more interesting relationships within the data. A software package from Universidad de La Rioja's EDMANS Groups was used to pre-process and analyze the time series from the NOAA datasets. The software package is called *KDSeries* and was created using *R*, a language for statistical computing.

The *KDSeries* package contains several functions that pre-process the time series data so knowledge discovery becomes easier and more efficient. The first step in pre-processing is filtering. The time series are filtered using a sliding-window filter chosen by the user. The filters included in *KDSeries* are Gaussian, rectangular, maximum, minimum, median and a filter based on the Fast Fourier Transform. Important minimum and maximum points of the filtered time series are then identified. The optima are used to identify important episodes in the time series. The episodes include increasing, decreasing, horizontal and over, below, or between a user-defined threshold.

After simple and complex episodes are defined, each episode is view as an item to create a transactional database. Another R based software package, *arules*, makes this possible. *Arules* provide algorithms that seek out items that appear within a window of a width defined by the user. From there, temporal association rules are then extracted from the database.

email: pmorreal@kean.edu

IEEE Symposium on Large Data Analysis and Visualization
October 23 - 24, Providence, Rhode Island, USA
978-1-4673-0155-8/11/$26.00 ©2011 IEEE

The first algorithm being used to extract the temporal association rules is the Eclat algorithm. Eclat (Echivalence Class Clustering and Bottom-up Lattice Traversal) is an efficient algorithm that generates frequent item sets in a depth-first manner. Other algorithms such as Aproiri and FP-growth will then be used to extract association rules and compared and contrasted with each other. This work is ongoing.

3 METHODOLOGY

The data used is located on NOAA's FTP site in the form of .dly files. Each station has its own .dly file which is updated daily (if the station still collects data). Each .dly file has all the data that has ever been collected for that station. Whenever new data is added for a station, it is appended to the end of the current .dly file, which presents the problem that each file must be downloaded over again to keep our local database current. To obtain the data, a Java-based program was built that would download every file in the folder holding the .dly files. The Java program used the Apache Commons Net library to download the files from the NOAA FTP server.

After downloading all of the .dly files, the Java program opens an input stream to each of the downloaded files (one at a time). Each line of the .dly files contains a separate data record, so the Java program would read in each line and use it to form a MySQL "INSERT" statement that would be used to place data into a local database. At one point, space was exhausted on the local machine, and researchers had to upgrade the hard disk from ~200 GB to 256 GB to continue inserting data into the relational database.

Once all of the data was placed into the local database, a web interface was built that allowed users to search the dataset (Figure 1). The interface allows users to search by country, state (within the United States), date range, and values that are <, <=, >, >=, !=, or == to any chosen value. Because the dataset contains the value -9999 for any record that is invalid or was not collected, the web interface also has the option to exclude any -999 values from the results. The results are output with each line containing a different data result, and each result consisting of month, day, year, and data value.

Figure 1. Query screen of web interface to NOAA GHCD dataset.

For visualizing the data, the first step was to plot the location of each station in Google Earth. The NOAA FTP server also has a file that lists the longitude, latitude, and elevation of each station, so this information was placed into a separate table in our database. Next, a PHP script from the Google Earth website was customized to support queries to the database for the location of each station and then format the results in KML. This KML data is then loaded into the browser based version of Google Earth on the same webpage. A separate PHP script was built that allows the user to search for stations in the entire world, by country, or by state (if searching within the US) (Figure 2).

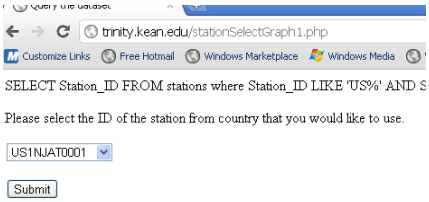

Figure 2. Web interface to station selection from GHCD dataset.

4 RESULTS

Results from the query are graphed (Figure 3).

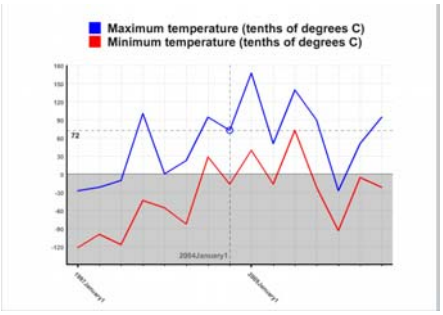

Figure 3. Real-time visualization of a user query to the GHCD dataset.

This visualization is also available for integration into Google Earth display (Figure 4).

Figure 4. NOAA GHCD reporting stations in Google Earth.

5 CONCLUSION

This work will migrate to Kean's CAVE, an immersive virtual reality environment, for visualization using Google Earth 3D.

REFERENCES

[1] A. Mueen and E. Keogh, "Online Discovery and Maintenance of Time Series Motifs", *Proceedings of 16th ACM Conference on Knowledge Discovery and Data Mining (KDD'10)*. pp. 1089-1098.

[2] P. Morreale, F. Qi, P. Croft, R. Suleski, B. Sinnicke, and F. Kendall. "Real-Time Environmental Monitoring and Notification for Public Safety." *IEEE Multimedia*, Vol. 17, No. 2, 2010, pp. 4-11.

[3] NOAA Global Historical Climatology Network (GHCN) Database http://www.ncdc.noaa.gov/oa/climate/ghcn-daily/

[4] NOAA Integrated Surface Database (ISD) http://www.ncdc.noaa.gov/oa/climate/isd/index.php