Efficient and interactive 3D visualization of large mined datasets

### Undergraduate Honours Project (COMP 4520) proposal prepared by Levko Ivanchuk, 7670173

## Project outline

Today, Big Data is a term commonly used to describe the vast amount of information harvested by big corporations & other institutions every day. This information is then used to provide deeper insights and strengthen the decisions made by such organizations. More and more companies’ employ frequent pattern mining to discover implicit, previously unknown and potentially useful knowledge from their data. Such knowledge is presented to the user in a form of frequent patterns or frequently occurring sets of items. Mostly, algorithms that find these frequent patters produce their results in a textual form, which is sometimes hard to comprehend and navigate. Thus users ability to extract useful knowledge and take decisions may be affected.

In this project, we aim to develop a visualization system that takes advantage of three-dimensional space and interactivity to improve comprehension and understanding of frequent pattern datasets. Our main contribution is the proposal and development of a 3 dimensional frequent pattern visualizer that can be scaled to huge datasets and remain highly interactive, while giving the user a lot of feedback of the frequency and support of the itemsets.

## Related work

Visualization of data mining results is at least as important as data mining itself. Mined data is of no use if people analyzing and interacting with it can not quickly understand what is going on and what the data is “telling” them. Researchers in areas of both data mining and visual analytics have looked at big data visualization for many years. Some examples include FIsViz[1], FpVAT[2], PowerSetViewer[3], Yang’s system[4][5]. We briefly discuss them in the remainder of this section.

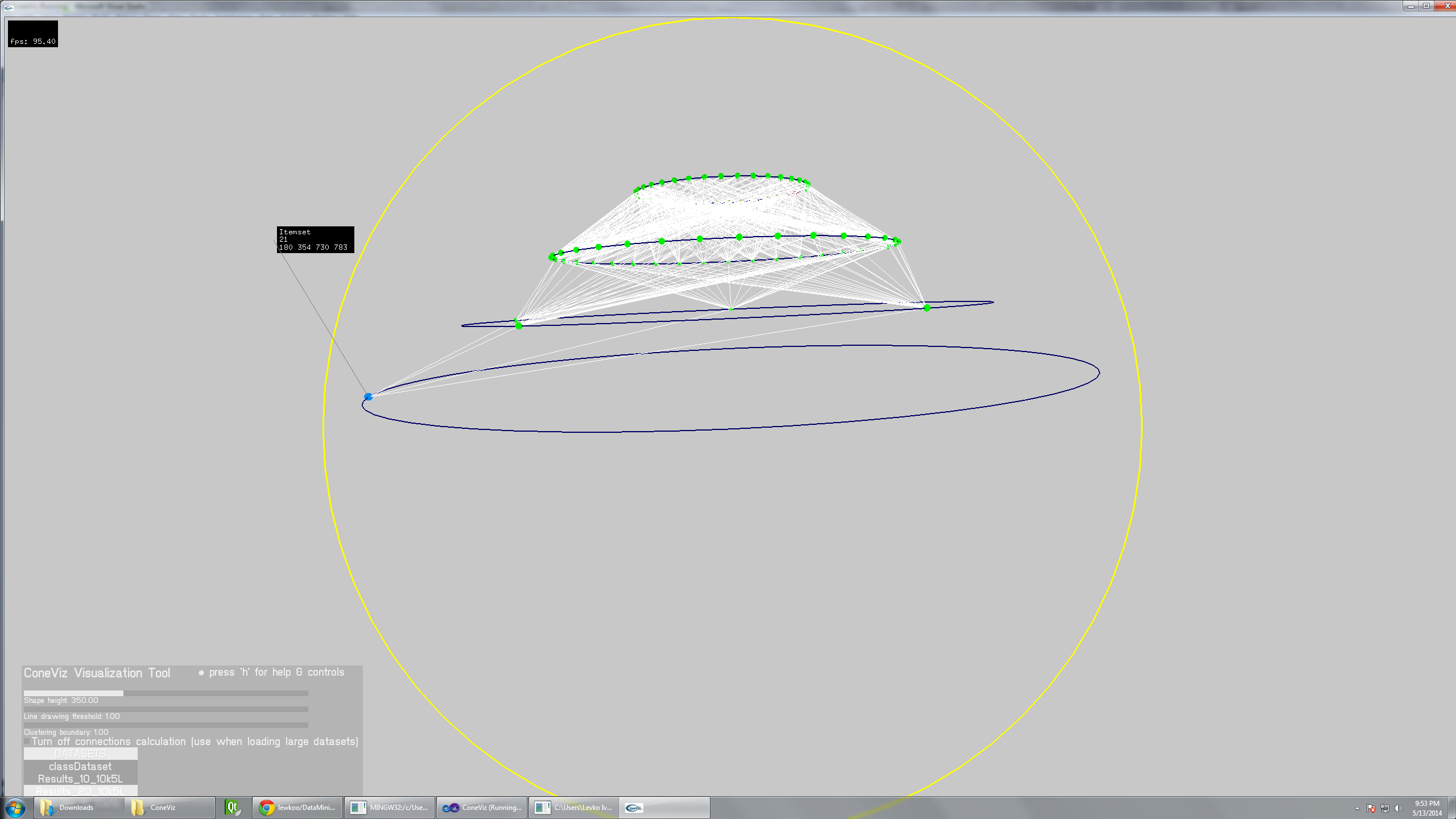
**FpVAT**

FpVAT is one of the recently developed frequent pattern visualizes. Consisting of two distinct modules, this system allows users to get an overview of the massive datasets, so that they can derive insights from it. Another module allows users to perform analytical reasoning via interactive visual interfaces to assist with detecting the expected frequent patterns and discovery of the unexpected ones.

The system uses a polyline method to connect the itemsets displayed on a 2 dimensional plane. As datasets grow larger, the number of lines that FpVAT displays will grow large, which in turn may create distractions to the user. In addition to that, it is not immediately obvious how to read the information that the system displays.

**PowerSetViewer**PowerSetViewer is a frequent patter visualizer that groups all patterns together based on cardinality and presents them in a two dimensional grid. It applies color coding to the background to indicate the cardinality of the itemsets. Whilst guaranteeing visibility, this visualization system also groups multiple patterns into the same square, which makes it hard to distinguish individual itemsets in the visualization. In addition, the system does not show the exact frequency of any given itemset.

## Research direction

****Previous work focused solely on two-dimensional visualization. Instead, we plan to take a new approach by adding an extra dimension. We believe that this approach can improve both learnability and scalability of the visualization by allowing for more interaction techniques (such as zooming in/out, rotating, flipping, spinning of the 3D model) and by simply allowing for more space in a 3D visualization. In addition, it allows for different geometrical shapes, color codings and other areas to experiment with.

This is a screenshot of the preliminary version of the system. Itemsets are represented as colored spheres, placed on circles (i.e. levels). The white lines represent the connections between the itemsets. Users also get a message box displayed next to their mouse cursor indicating the information about the selected itemset. By selecting an itemset, users are able to show only connections in the tree leading to that itemset.

**Project structure**

We see three main parts to this project:

* Data processing and shape generation
* Visualization load times & usability in terms of performance
* Interactivity of the visualization + the amount of information one can find while viewing

Some of the difficulties while working with large data sets involve processing times and memory consumption. Therefore, we propose to split the processing and shape generation from the viewing and interaction.

VIEWER

* accepts an meta-data augmented file + Polygon file for the shape itself
* allows user to interact with the visualization
* is optimized for viewing, not requiring the original data
* users are given extra features made possible by the metadata (sorting, searching, etc)

PARSER

- goes through the data

* provides non-interactive preview of the shape and appearance of the visualization
* allows designer to change some parameters & preview the final visualization
* generates an meta-data augmented Polygon file
* lots of parallel processing & optimizations

.PLY file + meta data

We propose an add-on for the mining process itself. By wiring it to the mining process, we can simultaneously generate the visualization data. Later, after the mining process is finished, one can preview the visualization as a screenshot, while also adjusting the parameters of the visualization (eg. shape height, frequency line threshold, clustering boundaries per level, etc.) and getting an almost instant rendering of the final visualization. The goal is to support at least 4 million records or higher. To do that we will employ some sort of parallel processing on the GPU (either shader or CUDA based), as most of the operations are highly adaptable to parallel processing.

Viewer will take the generated files from the parser and efficiently display them. Users will be allowed to interact with the shape, rotate it in any direction, as well as zoom in and zoom out. In addition, users will be able to select a given itemset and see all the connections leading to that particular itemset, Users will also see the exact frequency of the itemset, as well as all items that are in that selected set. Users should be allowed to filter by level, take a look at one level in isolation, drill down into the cluster, select an individual itemset even if the number of itemsets is very large. Also, a search feature will be useful. Again, some parallel processing might be needed here.

Additional directions of this project could include exploration of different 3D shapes that could potentially be useful at displaying frequent itemset mining results. Since there is a substantial amount of research on this topic in the areas of visual analytics, we might spend a bit more time reviewing the existing literature to find a suitable shape. In addition, we can cooperate with the HCI Lab at University of Manitoba in case we need to develop a better way to visualize and display frequent itemset mining results.

It would also be interesting to explore Unity for this project. Unity is a 3D graphics development suite that is becoming very popular and is increasingly being adopted by research community. It allows for quick and easy scripting of 3D shape & scene generation, as well as interaction with it. Using Unity will definitely add some novelty to this project and make it up-to-date with current 3D development techniques. However, using Unity might also be a limiting factor, as it is currently unknown to us how restricted the system it. Therefore, adopting Unity should be considered with caution.

## Facilities

Since the viewer will be aimed at a low-performance computer, there is not any extra computing facility required for this project. Frameworks and utilities that will be used must be open-source. The project should remain cross-platform, regardless of the chosen direction (i.e. Unity development or some other OpenGL frameworks).

## Anticipated length

Time wise, this project highly depends on the number of interactivity features that the system will support. Therefore, it will be important to develop an efficient and adaptable data parser, such that it can be modified to produce more metadata for the viewer, if needed. Since we already have some preliminary code for both data parser and the visualizer, and know where its weaknesses are, data parsing step should require no more than a month or 1.5 month of development. This leaves another 2 – 1.5 month to develop the viewer, evaluate it and wrap up the project by perhaps writing a short paper. Of course, these length estimations should be changed if we take the decision to use Unity for this project.

## Anticipated outcome

The main goal is to build a complete frequent pattern visualization system and evaluate if 3D visualization is of any potential benefit for visualizing this particular type of data over existing 2D visualizations. Such system should have high learnability and be easy to use. The goal is

Outline of the project's scope (abstract)

A background of the problem (literature review)

Research direction (proposed methodology)

Facilities (computing or other) required to do the research

An anticipated length of time requirement

Anticipated outcome

A list of relevant references