Visualizing Steam Game Information and Purchasing Records

Xuan Huang, Haihan Lin

Email, u1209767@utah.edu u12063262@utah.edu

uID, u1209767 u1206326

Github: https://github.com/xuanhuang1/vis-project

Background and Motivation

Both of us are conducting research in Graphics & visualizations, and the project starts as a common hobby of us. After exchanging ideas we are interested in how games as media are connected and influence players' behavior, as well as creative ways to visualize the result.

Steam also has its own hardware & software survey to present some of its user information, but as the title suggested the focus is on devices and operating systems only. There are also many third-party websites designed to provide insights into steam games data, such as Steamspy and SteamDB. These, however, often just displays and sort the data and thus serves mostly for ranking purpose.

Using two different data files, the customer data and the game description data, we are aimed at giving a new insight of the game dataset by connecting games through both tags and user behaviors.

Project Objectives

Given the two datasets, we discover that they are connected through game name. Purchase data provides individual steam customer behavior, while Steam Games data provides detailed information on each game sold on Steam. By connecting these, we can pose the following question: what other games that a user is more likely to purchase if they have purchased one game? How does each game connect to other games, and how would this connection affect a user's behavior? With this visualization, we want to answer these questions, and offer advertisement tips for game developers and market sales based on our findings.

While developing this project, there are several questions we need to answer:

1. The given dataset has too many attributes. What should be visualized to best answer the questions above?
2. Two dataset have only one attribute in common, which is the game name. What visualization design can best associate the two datasets?

Data

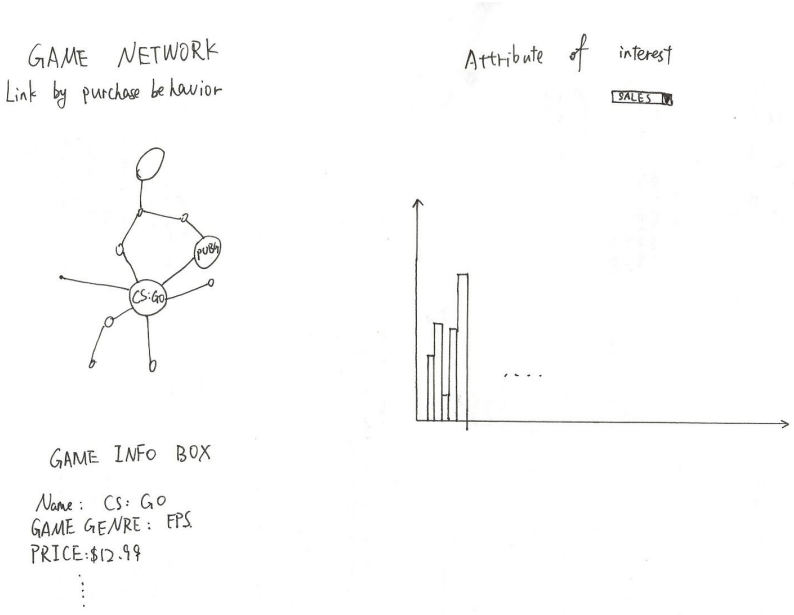
Both dataset comes from dataworld. The game-features.csv is adopted from a sample data project on github, and their original data is from public Steam datas on Steam's API and steamspy.com. The steam-200k-csv is from a kaggle page where they list 200k Steam user interaction.

Data Processing

The game-features dataset is a very large dataset and contains very extensive information about games on steam. It will slow the visualization significantly or even cause the browser not responding. Therefore we need to reduce columns in the dataset. Information such as minimal system requirement and supported languages is irrelevant to the question we are trying to answer. These columns will be deleted in data preprocessing. No extra quantities is expected to be derived from the data. The stream-200k-csv data only has four columns and are all essential to the visualization. So no cleanup will be done on this dataset.

Visualization Design

Design 1:



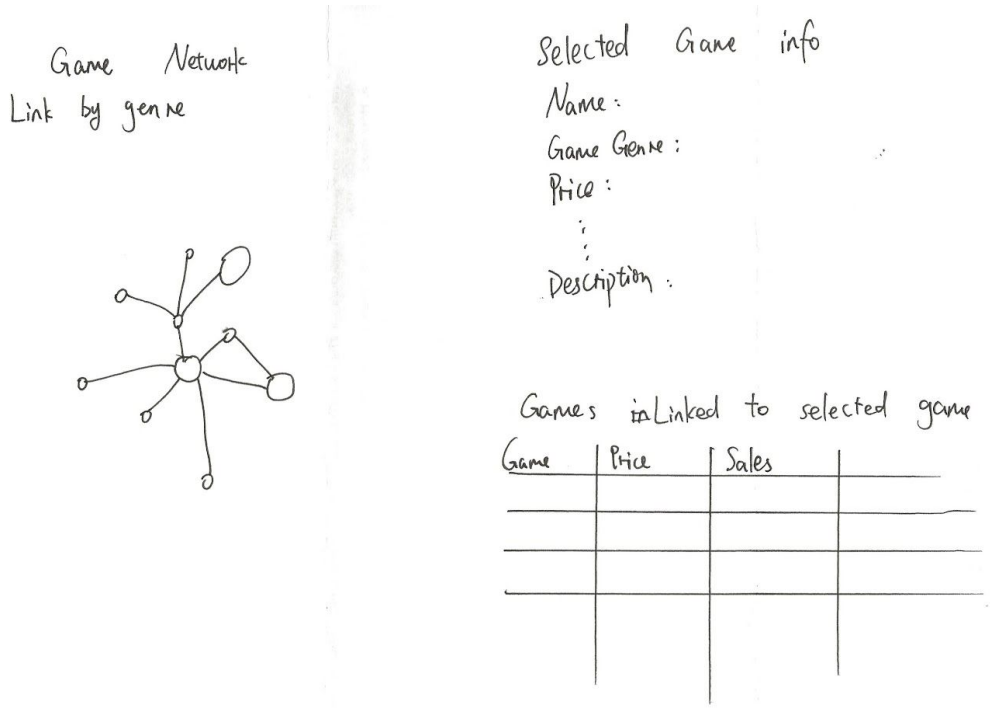
This is a multiview visualization. Left side is a network visualization. Nodes represent game titles, and there is a connection if the same user purchase the two games. The degree of node is the number of purchases on the same game. The network is interactive, and a user can select a game of interest. The info box below the network will show the detailed information of that game. On the

right side, it is a visualization of a attribute selected by the user. If a quantitative attribute is selected, it is visualized using bar chart. If a qualitative attribute is selected, it is visualized as a table, column as each quality input, and the game title as a row under the column if it has the quality.

To connect the network and the attribute visualization on the right, we use highlighting to visualize the selected game title. If a node is selected in the network, the associated bar or title will be highlighted, and vice versa.

This visualization helps us to understand how a single attribute would affect the user's purchase behavior, and learn what has a bigger influence on user's decision.

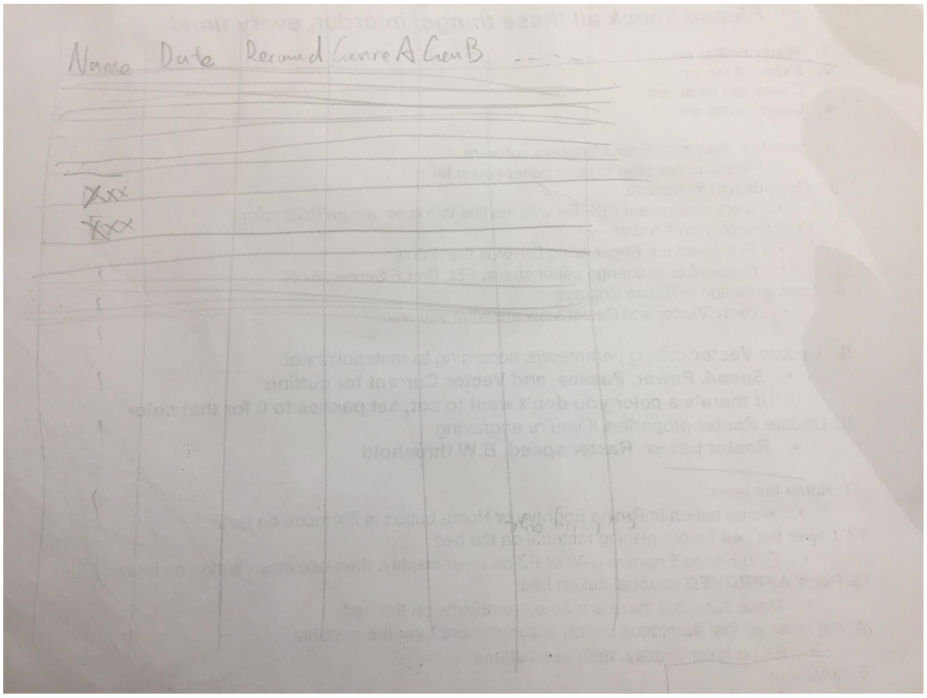
Design 2:



This design changes how the network is based on. Games are represented as nodes, and there is a link between two nodes if they share the same genre. And the degree of node is the number of purchases of a game. Similar to Design 1, it uses a info panel to show the detailed information if a game is selected. Instead of showing an attribute of interest, this shows all the attributes of the games being selected and its neighbors in the table. And the table can be sorted based on attribute selected.

This design is very easy to compare among all attributes, which makes it very easy to compare attributes in the table, and see if games that are connected have patterns among the attributes presented.

Design 3:



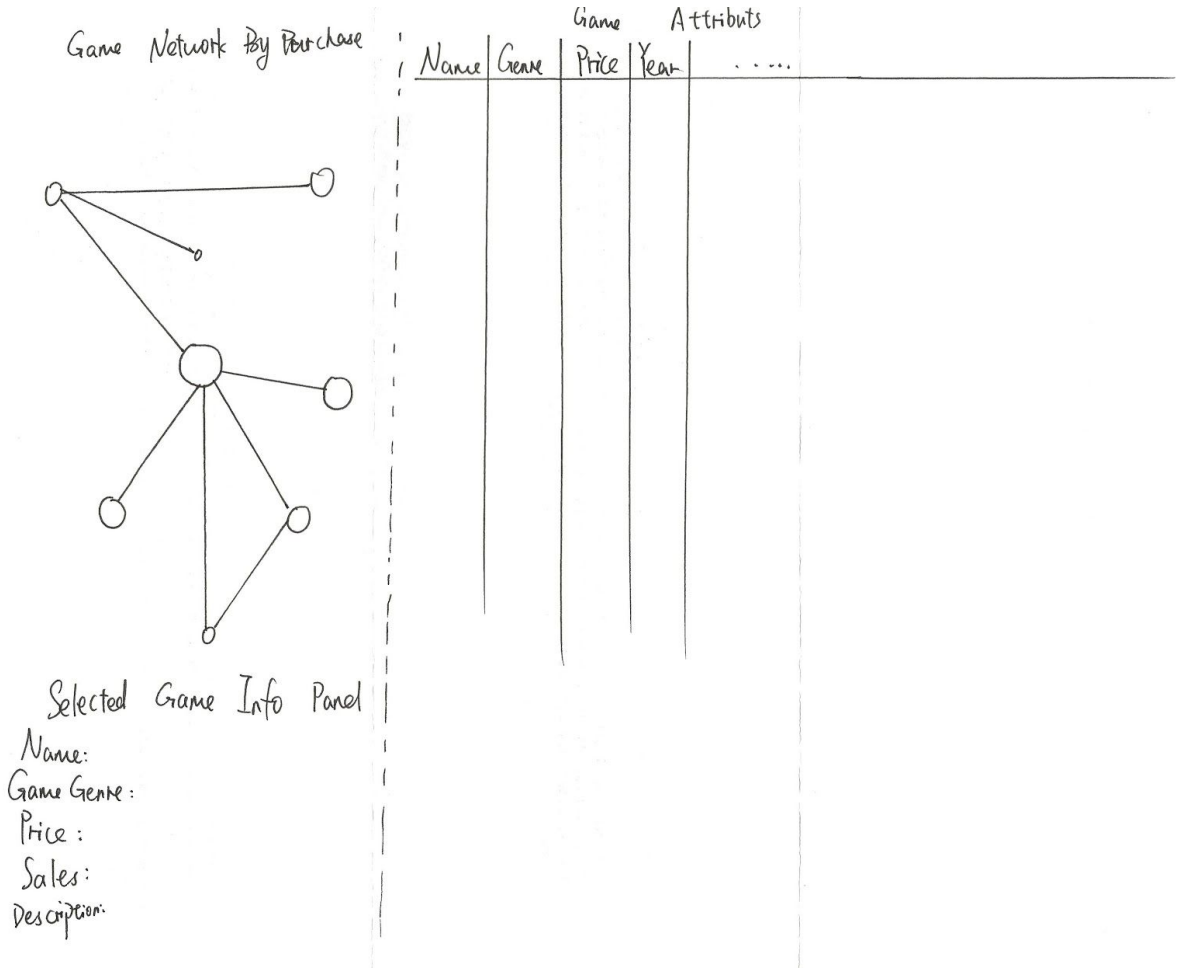
We can incorporate the idea of a magic lense into the game feature chart. The chart will be loaded in as the original, plus a column for the score from our relationship evaluation function. The function is generated by the games-200k-csv file according to user behaviors, and will be displayed and modifiable.

The chart will be resorted according to our function output. Only the selected row and its surrounding area are shown. And users can click to expand other areas of the chart to view details.

Others will be collapsed with only the genres values (true or false in the chart) shown as a black line for true or nothing for false.

Here we provide a quick 1D view of relationships, yet all information about the games are available when requested.

Final Design



The final design includes the network of games in design 1, and the attributes table in design 3. The network is based on users' purchasing behavior. The right table shows all the available data initially. Qualitative data is encoded with the the info written in the table directly, or simple dots showing categories, Quantitative data is encoded with a bar in each row. The table can be sorted according one's preference, and filtered based on attributes(only show rows with certain attributes). When selected in the network, the table will be updated to only show rows that are selected game and its neighbor. And when selected in the table, the network will only show nodes that contains the selected game and its neighbor in the network.

Must-Have Features

1. Displaying a list of the game dataset of size 10k
2. Select one game or a subset of games
3. Group (or connect in someway) games according to genre tags
4. Sort/rank the games according to user behavior data
5. An info panel to show detail attributes & descriptions
6. Multi-view visualizations

Optional Features

1. 3D view if appropriate
2. Display all games in the dataset (match all names and incorporate DLCs correctly)
3. Search for a particular game from user text input
4. System dealing with attributes other than specified above (prices, languages, number of packages etc)

Project Schedule

Week 1: clean data, decide data structure in js and view layout in html

Week 2: Populate data into webpage, has a working prototype

Week 3: Implement must have features

Week 4: Implement optional features if the progress allows