Visualizing Clickstream Data as Discrete Time Markov Chains

Shirbi Ish-Shalom, Samuel Hansen

CS448B: Data Visualization, Spring 2016, Stanford University

Abstract

From eCommerce to online dating, understanding how users navigate web pages is vital to online businesses. When a user visits a web page, she generates a sequence of page visits, known as a "clickstream". Taken together, aggregated clickstreams can answer many important questions, including how well a site predict page transitions, facilitate interpage traffic, and identify behavioral cohorts of users.

To assist the analysis of clickstream data, we present *Clickstream Explorer*, a visualization dashboard that represents aggregate clickstream data as a discrete time Markov chain. *Clickstream Explorer* improves upon prior visualization tools by employing:

1) Integration with the R computing language;
2) Multiple display options (graph view, table view, heatmap view, etc.);
3) Dynamic updating via user inputs.

In doing so, *Clickstream Explorer* enables the rapid exploration of clickstream data and associated Markov chain properties.

Motivation

Prior work such as *WebQuilt*[12], depicted in Figure 1, and *StatViz*[7] have represented clickstreams as hierarchical trees, in which nodes represent webpages, links represent transitions, and tree depth represents click number.

Fig.1: *WebQuilt* Clickstream Visualization Software.[12]

However, such tools become difficult to interpret as tree depth increases, prompting researchers to adopt the Markov assumption: given a certain state, no additional information is needed to predict the next state,[1,4,8,11] expressed formally:

$$\mathbb{P}(X_n = x_n \mid X_n = x_{n-1}, \ldots, X_0 = x_0) = \mathbb{P}(X_n = x_n \mid X_n = x_{n-1})$$

Although researchers have used this Markov property to develop models of web browsing,[1,6,11] build statistical analysis packages for clickstream data[3,9,10] and predict user buying patterns,[5] few tools provide comprehensive visualizations of Markov chain representations of clickstream data. Current Markov visualization tools in R - the most widely used statistical language - suffer from occlusion and lack of scalability. To solve this problem, we built an interactive dashboard to minimize issues of occlusion, display Markov chain properties, and facilitate rapid comprehension of aggregate clickstream behavior.

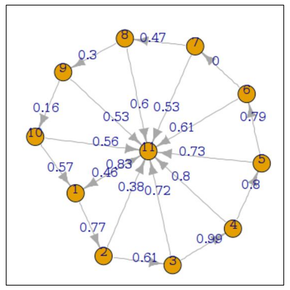


Fig. 2: Example Markov chain visualization from the *markovChain* package in R.

Methods

We created our dashboard in the following 6 steps:

1 Clean Data

We obtained 989,818 clickstreams from *msnbc.com* that were collected on Sept. 28, 1999. Each URL was pre-categorized into 17 page categories (i.e. “News”, “Weather”, “Sports”). Each clickstream was a comma-separated string, represented:

Frontpage, Sports, Sports, Sports
Frontpage, Tech
News, Travel, Weather

We cleaned the data by filtering out observations with only one page visit because we wanted to visualize clickstreams that had at least one page transition.

2 Convert to Transition Matrix

To convert our clickstreams into an aggregate transition matrix, we first computed a matrix with bigram counts from each state *i* to each state *j*. Then, we normalized this matrix by each row sum to compute transition probabilities.

Bigram Count Matrix

	Frontpage	News	Tech
Frontpage	6	3	1
News	3	3	4
Tech	1	1	8

Normalize

Transition Matrix

	Frontpage	News	Tech
Frontpage	0.6	0.3	0.1
News	0.3	0.3	0.4
Tech	0.1	0.1	0.8

3 Compute Markov Chain Properties

We created a pipeline to analyze the following properties of the Markov chain defined by our computed transition matrix:

1) **Irreducibility**: Are all states reachable from all other states?
2) **Periodicity**: Do cycles exist in the chain?
3) **Invariant Distribution**: Does a unique invariant distribution exist?

For instance, our pipeline would classify the chain in Fig. 3 as non-irreducible (because there are absorbing states), aperiodic (because self-loops exist), and having no unique invariant distribution (given the *Convergence Theorem*).

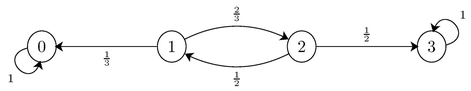


Fig. 3: Example of a non-irreducible, aperiodic Markov chain.

4 Create Weighted Edge List

Using the transition matrix, we created a weighted edge list, which included the source page, target page, and respective transition probability as the edge weight. We used this edge list to create several visualizations, including our directed graph and heatmap.

Edge List

Source	Target	Weight
Frontpage	Frontpage	0.6
Frontpage	News	0.3
Frontpage	Tech	0.1

5 Build RShiny Dashboard UI

Because many Markov chain analysis tools exist in R, we wanted to integrate our visualization tool into pre-existing frameworks. This would allow statistical programmers who employ R packages such as *markovChain, DTMC*, and others to easily jump between the RStudio command line and our *Clickstream Explorer.*

In turn, we built an RShiny Dashboard application that can be run directly from the RStudio command line. As a first step, we built the user interface (UI) component, which included sliders, toggle buttons, tabs, and other interactive widgets.

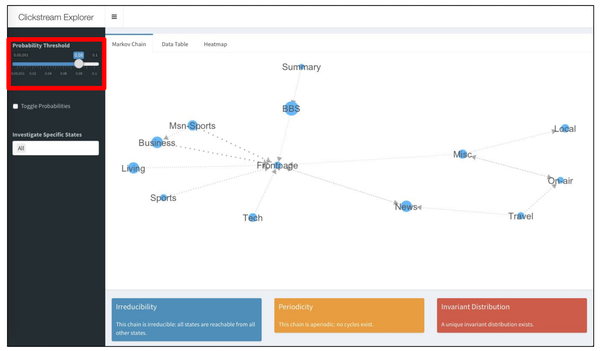
6 Connect RShiny Dashboard Server

As a second step, we connected our UI to a separate server script, which dynamically updates graphical displays to be rendered on the dashboard. We used the computed edge list and Markov chain properties to build customized Markov graphs, interactive data tables, and heatmaps.

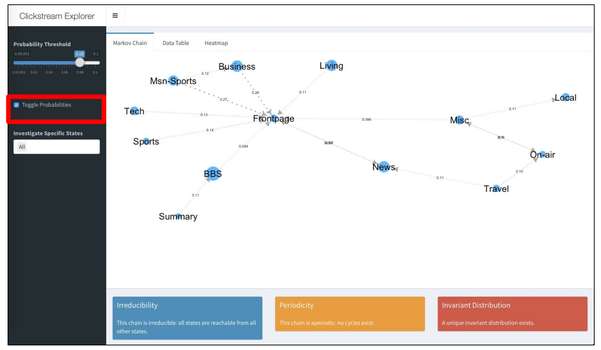
Results

The dashboard implements the following functionalities:

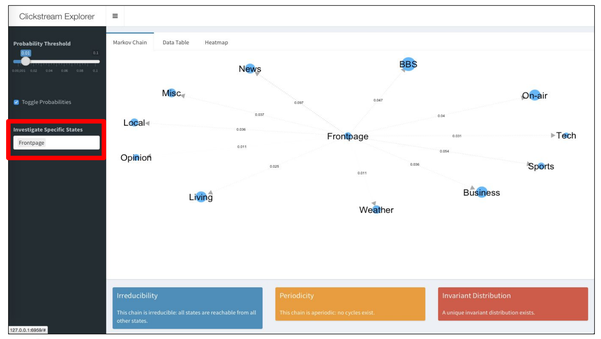
1 Filter edges by transition probability



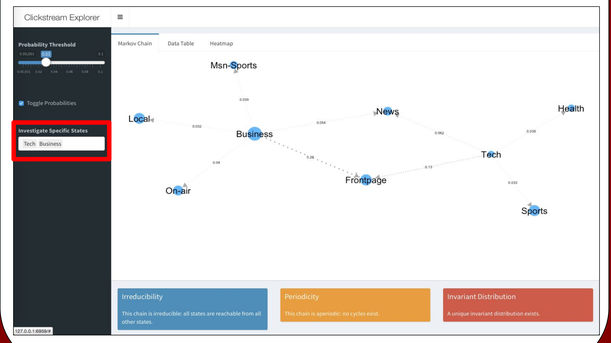
2 Toggle probability labels ON/OFF



3 Select individual source node



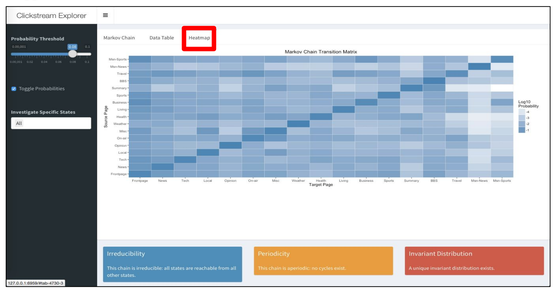
4 Select multiple source nodes



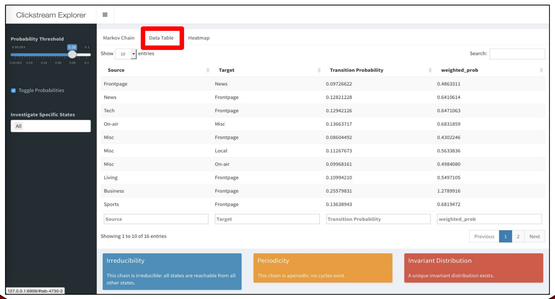
Results

The dashboard also includes two additional tab views:

5 View transition matrix heatmap



6 Search nodes in data table



Future Work

1. Automate Classification of URLs to Page Categories

Markov chains representations of clickstreams are most effective when the state space is relatively small (e.g., less than 50). However, treating each unique URL as its own state leads to enormous state spaces, implying URLs must be classified into more general categories. Future work should pursue strategies to classify URLs into Markov state categories (e.g., via similarity-based clustering techniques).

2. Implement Markov-Property Validation Tool

We selected our dataset partly because prior empirical testing demonstrated first-order Markov models are suitable to approximate aggregate clickstream behavior on *msnbc.com*.[2] However, to generalize to future datasets, future work could develop a pipeline to validate whether a given dataset satisfies the first-order Markov assumption. Such a validation tool could be integrated into *Clickstream Explorer,* in addition to a dataset upload feature.

References

[1] Bucklin, Randolph E, and Catarina Sismeiro. “A Model of Web Site Browsing Behavior Estimated on Clickstream Data.” *Journal of Marketing Research* XL.August (2003): 249–267. Web.
[2] Cadez, Igor et al. “Visualization of Navigation Patterns on a Web Site Using Model Based Clustering.” *Data Mining and Knowledge Discovery* 7.4 (2000): 399–424. Web.
[3] Hahsler, M, and M H Dunham. “rEMM: Extensible Markov Model for Data Stream Clustering in R.” *Journal of Statistical Software* 35.5 (2010): 1–31. Web.
[4] Lakshminarayan, Choudur, Ram Kosuru, and Meichen Hsu. “Modeling Complex Clickstream Data by Stochastic Models : Theory and Methods.” *WWW '16 Companion*. Geneva, Switzerland: N.p., 2016. 879–884. Web.
[5] Lim, Miseon, Hyunsoo Byun, and Jinhwa Kim. “A Web Usage Mining for Modeling Buying Behavior at a Web Store Using Network Analysis.” 8.October (2015): n. pag. Web.
[6] Montgomery, Alan L et al. “Modeling Online Browsing and Path Analysis Using Clickstream Data.” *Marketing Science* 23.4 (2004): 579–595. Web.
[7] Pinstein, Alan. “StatViz.” 2013. Web.
[8] Sadagopan, Narayanan, and Jie Li. “Characterizing Typical and Atypical User Sessions in Clickstreams.” *Proceedings of the 17th international conference on …* (2008): 885. Web.
[9] Scholz, Michael. “Package ‘ Clickstream .’” 2016. Web.
[10] Spedicato, Giorgio Alfredo, and Mirko Signorelli. “The R Package ``markovchain'': Easily Handling Discrete Markov Chains in R.” *Cran* (2014): Web.
[11] Wang, Gang et al. “You Are How You Click : Clickstream Analysis for Sybil Detection.” (2013): Web.
[12] Waterson, Sarah J et al. “What Did They Do? Understanding Clickstreams with the WebQuilt Visualization System.” *Proceedings of the Working Conference on Advanced Visual Interfaces AVI 02* (2002): 94. Web.