Data-Driven Interactions for Web Design

Ranjitha Kumar
Department of Computer Science
Stanford University
ranju@cs.stanford.edu

ABSTRACT

This thesis describes how data-driven approaches to Web design problems can enable useful interactions for designers. It presents three machine learning applications which enable new interaction mechanisms for Web design: rapid retargeting between page designs, scalable design search, and generative probabilistic model induction to support design interactions cast as probabilistic inference. It also presents a scalable architecture for efficient data-mining on Web designs, which supports these three applications.

Author Keywords

Web design; machine learning.

ACM Classification Keywords

H.1.2 [Models and Principles]: User/Machine Systems – Human factors.

General Terms

Algorithms; Design; Human Factors.

MACHINE LEARNING FOR WEB DESIGN

The Web provides an enormous repository of design knowledge: every page represents a concrete example of human creativity and aesthetics. Given the ready availability of Web data, how can we leverage it to help designers?

This thesis describes three machine learning applications which enable new interaction mechanisms for Web design: rapid retargeting between page designs to automatically transfer the content from one page into the style and layout of another, scalable design search for finding relevant examples during ideation and implementation, and inducing generative probabilistic models from exemplars to support interactions cast as probabilistic inference. It also describes a common infrastructure used in all three applications to support large-scale data-mining and machine learning on Web designs.

There are three major takeaways from the thesis:

1. *Data-driven approaches can enable new, principled design interactions that allow people to work with examples in a natural way.* Leveraging examples of previous work is an established technique in design, but current practices for working with them are largely informal [9, 3]. Machine learning applications can establish new workflows that allow people to more directly use and learn from the design knowledge on the Web.

2. *Leveraging structure that is intrinsic to Web designs is key to building more powerful design interactions.* On the Web, every page is associated with a Document Object Model (DOM) tree, which can be used along with render-time information to bootstrap a visual information hierarchy for designs. Previous work has looked at statistical profiles of Web pages to aid designers, but treat pages as a flat collection of features and ignore the underlying structure [5, 15]. This thesis demonstrates that structure is the key to enabling powerful design interactions.

3. *Understanding the way people think about design can inform the construction of learning algorithms.* Prior systems that facilitate work with examples use *ad hoc* heuristics to reason about page designs [8, 10]. A more principled approach is to learn these rules directly from crowdsourced data. The learning applications presented in this thesis train on data collected from people.

Figure 1. Our example-based retargeting algorithm, *Bricolage*, automatically renders pages in new layouts and styles.

UIST'12, October 7–10, 2012, Cambridge, Massachusetts, USA.
ACM 978-1-4503-1582-1/12/10.

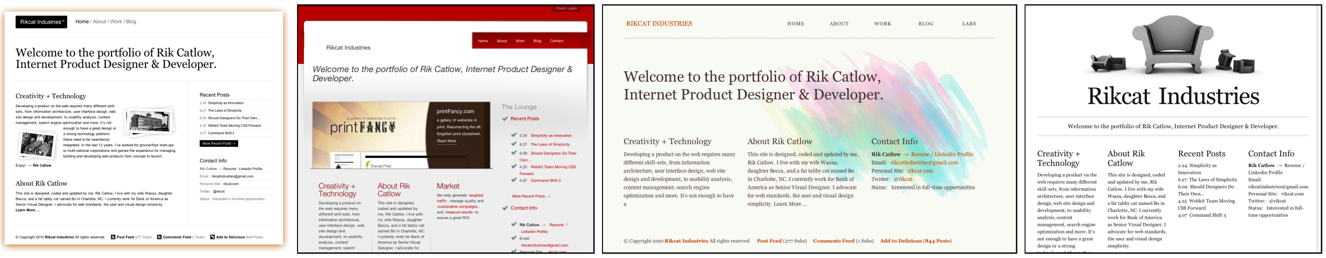


Figure 2. Flexible tree matching used to rapidly prototype many alternatives. *Left:* the original Web page. *Rest:* the page automatically retargeted to three other layouts and styles.

RAPID RETARGETING BETWEEN PAGE DESIGNS

People frequently rely on templates when designing Web sites. While templates provide a simple mechanism for rendering content in different layouts, their rigidity often limits customization and yields cookie-cutter designs. This thesis presents *Bricolage*, a structured prediction algorithm that allows any page on the Web to serve as a design template [6]. The algorithm works by matching visually and semantically similar elements in pages to create coherent mappings between them. Once constructed, these mappings are used to automatically transfer the content from one page into the style and layout of another (Figure 1).

Bricolage uses structured prediction [2] to learn how to transfer content between pages. It trains on a corpus of human-generated mappings, collected using a Web-based crowdsourcing interface. The mapping collector was seeded with 50 popular Web pages that were decomposed into a visual hierarchy by our constraint-based page segmentation algorithm, Bento. In an online study, 39 participants with some Web design experience specified correspondences between page regions and answered free-response questions about their rationale.

These mappings guided the design of Bricolage's matching algorithm. We found consistent structural patterns in how people created mappings between pages. Participants not only identified elements with similar visual and semantic properties, but also used their location in the pages' hierarchies to guide their assignments. Consequently, Bricolage employs a novel tree-matching algorithm that balances visual, semantic, and structural considerations [7].

To test the effectiveness of Bricolage's machine learning components, we ran a hold-out test. Bricolage is able to reproduce human mappings with nearly 80% accuracy. Moreover, we show that flexibly preserving structure is essential for predicting human-like mappings. If we don't account for structure, the accuracy drops to 53%, substantially decreasing the predictive power of the algorithm. This algorithm enables a diverse set of design interactions, including rapid prototyping (Figure 2), retargeting between form factors (Figure 3), and measuring the similarity of Web designs.

SCALABLE DESIGN SEARCH

Designers leverage examples during ideation to understand the space of possible designs and learn implementation techniques [1]. However, modern search engines offer little support for design-based search, making it difficult to find relevant examples.

Furthermore, search is only useful if it can be deployed on a truly large scale: in a database of a few hundred pages, the likelihood that a designer will find a useful example is fairly low [10]; in a database of ten million pages, the likelihood increases. Text-based search engines process queries efficiently by computing bag-of-words representations of documents; no such natural vector space describes page designs. Pages are not so obviously fixed-dimensional: they have varying numbers of elements and topology. How do we express *topology* as a *fixed-dimensional* vector to afford fast comparisons?

This thesis demonstrates that a meaningful search space can be constructed via deep learning, using recent work on recursive neural networks (RNNs) to induce a fixed-dimensional, structurally-sensitive embedding for each element in a page's visual hierarchy [11]. The RNN framework leverages a set of canonical features to bootstrap a continuous vector space representation for each variable-sized region in a page.

The key idea behind RNNs is that the vector representation for a node is a function of its children. Given a node $\mathbf{p_1}$ and its children $\mathbf{c_1}$ and $\mathbf{c_2}$, $\mathbf{p_1} = f(\mathbf{W}[\mathbf{c_1}\ \ \mathbf{c_2}]^T)$. In classical neural networks, the transformation matrix, $\mathbf{W}$, would vary across the hierarchy, but in RNNs the same matrix is used to compute the activation at every node, embedding every node in the same space. This formulation accounts for the recursive structure of page hierarchies, where similar nodes may exist at different scales.

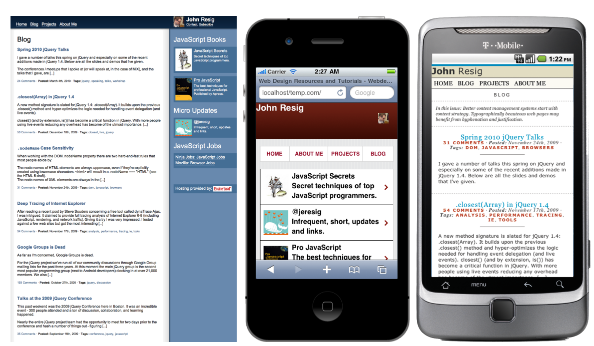


Figure 3. Flexible tree matching can be used to retarget Web pages designed for the desktop to mobile devices. *Left:* the original Web page. *Right:* the page automatically retargeted to two different mobile layouts.

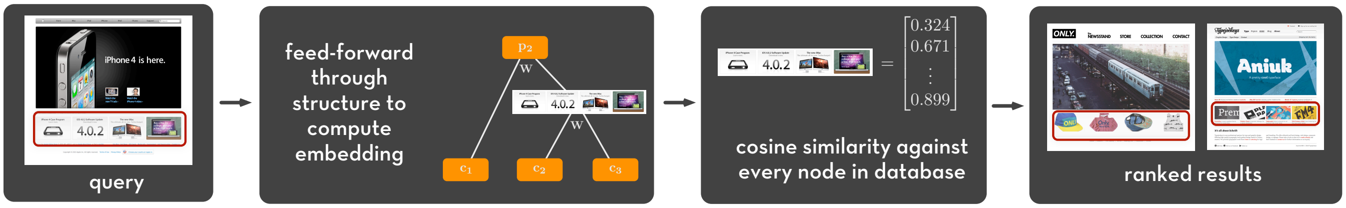


Figure 4. With these fixed-dimensional representations, we can use standard IR techniques to search for designs.

Given this RNN framework, we can turn to people to define what constitutes similarity between pages and page elements. We want to train the parameters of the RNN (*e.g.* $\mathbf{W}$) so that similar pages and page elements have vector representations that are close together. We can collect two types of labels through a Web-based crowdsourcing interface: stylistic labels (*e.g.* minimal, fun) that describe the entire page design and structural semantic labels (*e.g.* header, logo) that describe page elements in the hierarchy. Then, the training corpus consists of page hierarchies where certain nodes are augmented with a softmax layer and label. Training works via backpropogation through structure: the goal is to iteratively adjust the parameters to minimize the error function, which is the sum of squared differences between the assigned and predicted labels.

By using this representation in a standard cosine similarity framework, we can enable several different types of search queries. Users can select a page, and ask to see similar pages in the database. Since every node in a page has its own embedding, users can also search at multiple scales: searching for similar page elements is the same operation as searching for a similar page (Figure 4). Moreover, the system can predict text-based labels across the corpus, allowing users to perform keyword searches on stylistic and structural semantic labels.

GENERATIVE PROBABILISTIC MODELS FOR DESIGN

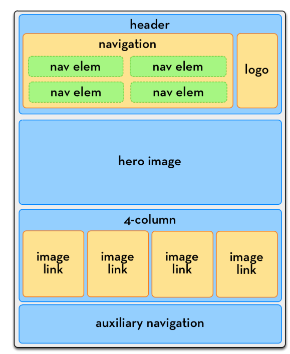
When building sites, skilled designers often rely on formalized knowledge about design patterns, typically encapsulated in books or style guides [14]. Such rules for good design, however, are difficult to enumerate and operationalize. A more attractive proposition is to learn these rules from data.

To learn patterns in a principled way, we leverage techniques from natural language processing and structured concept learning [4]. In particular, we cast the problem as *grammar induction*: given a corpus of example designs, we induce a probabilistic formal grammar over the exemplars. Once learned, this grammar gives a design pattern in a human-readable form that can be used to synthesize novel designs and verify extant constructions [13].

The crux of this induction is learning how to generalize beyond the set of exemplars: we would like to distill general principles from the provided designs without extrapolating patterns that are not supported by the data. In this thesis, we describe how to use an iterative structure learning technique called Bayesian Model Merging [12] to learn design patterns. The method formulates grammar induction as Bayesian inference and employs an inductive bias based on the law of succinctness, also known as Occam's razor, searching for the simplest grammar that best fits the examples. Since compactness and generality are inexorably linked in grammar-based models, the method provides a data-driven way to learn design patterns that are neither too specific nor overly general.

Using this Bayesian Model Merging technique, we induced a grammar on Web designs from a set of hand-labeled Web page hierarchies. Figure 5 shows a small portion of our corpus of Web pages, as well as a few random derivations from the learned model of page structures.

Although this is just a first step towards learning design patterns from data in a principled way, this thesis discusses the potential for exploiting the rich mathematical structure of generative probabilistic models in tool-building. In particular, we describe how some common design tasks can be formulated as probabilistic inference problems over a particular design pattern, and solved with Monte Carlo methods. For instance, we can write down a smooth function that scores page designs based on how closely they conform to a particular specification, and perform MAP estimation via MCMC to find a page from our learned Web grammar that has a four-column layout, a single navigation bar with four navigation elements, a header, and a hero image (see inset).



DATA MINING ON DESIGNS

Machine learning applications trained on page designs require efficient access to the design information from a large corpus of Web pages. To access the design information of a Web page, it is not sufficient to examine its raw HTML: the page must be rendered. Although traditional Web crawlers make it easy to scrape content from pages, acquiring and managing all the resources necessary to preserve a page's render-time appearance is much more difficult. Furthermore, with the advent of client- and server-side scripting and dynamic HTML, many modern Web pages are mutable and may change between accesses, frustrating algorithms that expect consistent training data.

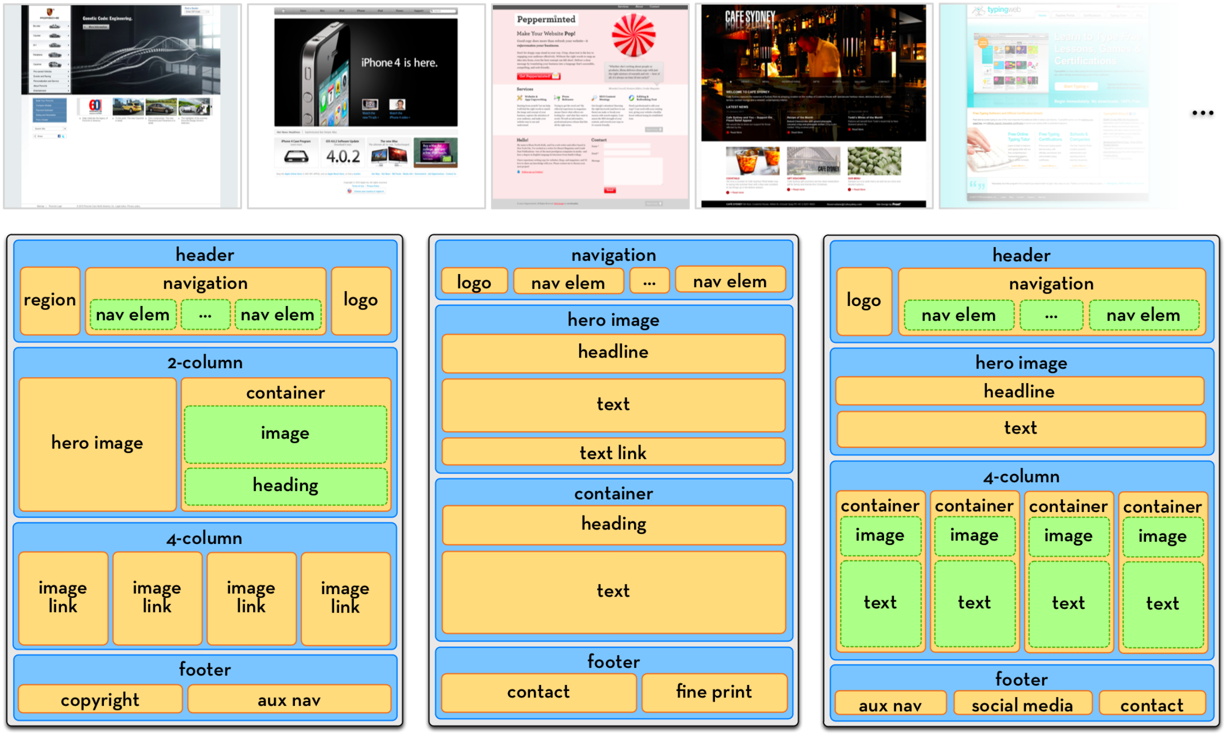


Figure 5. (top) A small sampling of the Web pages in our corpus. (bottom) Three random derivations from the grammar induced over that corpus.

To overcome these challenges, this thesis describes a new kind of Web repository. The repository is populated via a bespoke Web crawler, which requests pages through a caching proxy backed by an SQL database. As a page is crawled, all requested resources are versioned and stored, its DOM tree is processed to produce a static visual hierarchy of the page's structure, and a set of semantic and vision-based features are calculated on each constituent page component. These structures are then exposed through a RESTful API, allowing fast component-wise queries on features. We have found that this design repository enables the rapid development of a diverse set of machine learning applications that support creative work.

FUTURE WORK

This thesis focuses on understanding how data-driven approaches can aid Web designers. The Web holds appeal as an anchoring domain for three major reasons: the scale and diversity of harvestable examples; the popularity and utility of Web design for both novices and experts; and the overall impact of the Web on everyday life. However, it seems likely that these techniques could prove useful in other design domains as well. Bringing data into any design endeavor could amplify human creativity.

ACKNOWLEDGMENTS

The author would like to thank Scott Klemmer and all of her collaborators. This work has been supported by a Google PhD Fellowship and NSF Grant IIS-0745320.

REFERENCES

1. Buxton, B. *Sketching User Experiences*. Morgan Kaufmann, 2007.
2. Collins, M. Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms. In *Proc. EMNLP* (2002).
3. Herring, S. R., Chang, C.-C., Krantzler, J., and Bailey, B. P. Getting inspired!: understanding how and why examples are used in creative design practice. In *Proc. CHI* (2009).
4. Hwang, I., Stuhlmüller, A., and Goodman, N. D. Inducing probabilistic programs by Bayesian program merging. *CoRR abs/1110.5667* (2011).
5. Ivory, M. Y., and Hearst, M. A. Statistical profiles of highly-rated Web sites. In *Proc. CHI* (2002).
6. Kumar, R., Talton, J. O., Ahmad, S., and Klemmer, S. R. Bricolage: Example-based retargeting for web design. In *Proc. CHI*, ACM (2011).
7. Kumar, R., Talton, J. O., Ahmad, S., Roughgarden, T., and Klemmer, S. R. Flexible tree matching. In *Proc. IJCAI* (2011).
8. Lee, B., Srivastava, S., Kumar, R., Brafman, R., and Klemmer, S. R. Designing with interactive example galleries. In *Proc. CHI* (2010).
9. Newman, M. W., and Landay, J. A. Sitemaps, storyboards, and specifications: a sketch of Web site design practice. In *Proc. DIS* (2000).
10. Ritchie, D., Kejriwal, A., and Klemmer, S. R. d. tour: style-based exploration of design example galleries. In *Proc. UIST* (2011).
11. Socher, R., Lin, C. C., Ng, A. Y., and Manning, C. D. Parsing natural scenes and natural language with recursive neural networks. In *Proc. ICML* (2011).
12. Stolcke, A., and Omohundro, S. M. Inducing probabilistic grammars by Bayesian model merging. In *Proc. International Colloquium on Grammatical Inference and Applications* (1994).
13. Talton, J., Yang, L., Kumar, R., Lim, M., Goodman, N. D., and Měch, R. Learning design patterns with bayesian grammar induction. In *Proc. UIST* (2012).
14. van Duyne, D. K., Landay, J. A., and Hong, J. I. *The Design of Sites*. Addison-Wesley, 2002.
15. Zheng, X. S., Chakraborty, I., Lin, J. J.-W., and Rauschenberger, R. Correlating low-level image statistics with users - rapid aesthetic and affective judgments of web pages. In *Proc. CHI* (2009).