Data-driven Web Design

Ranjitha Kumar RANJU@CS.STANFORD.EDU
Stanford University, Stanford, CA

Jerry O. Talton JERRY.O.TALTON@INTEL.COM
Intel Corporation, Hillsboro, OR

Salman Ahmad SAAHMAD@MIT.EDU
Massachusetts Institute of Technology, Cambridge, MA

Scott R. Klemmer SRK@CS.STANFORD.EDU
Stanford University, Stanford, CA

Abstract

This short paper summarizes challenges and opportunities of applying machine learning methods to Web design problems, and describes how structured prediction, deep learning, and probabilistic program induction can enable useful interactions for designers. We intend for these techniques to foster new work in data-driven Web design.

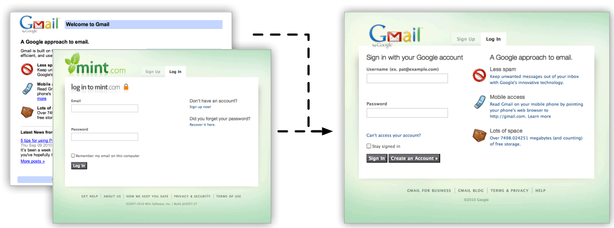

Figure 1. Our example-based retargeting algorithm, *Bricolage*, automatically renders pages in new layouts and styles.

1. 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 note outlines three machine learning applications which enable new interaction mechanisms for Web design: structured prediction for rapid retargeting, deep learning for design-based search, and probabilistic program induction for operationalizing design patterns.

All of these techniques leverage *structure* that is intrinsic to Web designs. In machine learning applications, working with structured representations affords significant advantages over unstructured data sets, such as images or text (Socher et al., 2011). 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.

To apply machine learning techniques to Web design, we need to collect a corpus of training examples. 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.

To overcome these challenges, we have constructed 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

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

have found that this design repository enables the rapid development of a diverse set of machine learning applications that support creative work.

2. Structured Prediction

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. *Bricolage* is a structured prediction algorithm that allows any page on the Web to serve as a design template (Kumar et al., 2011a). 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 learns correspondences between pages by training on a set of human mappings gathered via crowdsourcing. It learns how to flexibly optimize visual, semantic, and structural considerations using the generalized perceptron algorithm (Kumar et al., 2011b). Our experiments show that flexibly preserving structure is essential for predicting human-like mappings. The method enables a diverse set of design interactions, including rapid prototyping, retargeting between form factors, and measuring the similarity of Web designs.

3. Deep Learning

Designers leverage examples during ideation to understand the space of possible designs and learn implementation techniques (Buxton, 2007). However, modern search engines offer little support for design-based search, making it difficult to find relevant examples. Text-based search engines process queries efficiently by computing bag-of-words representations of documents; no such natural vector space describes page designs. Instead, we can construct a meaningful search space 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 (Socher et al., 2011). The RNN framework leverages a set of canonical features to bootstrap a continuous vector space representation for each variable-sized region in a page.

By using this representation in a standard cosine similarity framework, users could "query-by-part" to search for pages containing similar design elements. Moreover, by augmenting each node in the RNN with softmax layers, the system could train text-based labels across the corpus, allowing users to perform keyword searches on stylistic (*e.g.* minimal, fun) or structural (*e.g.* header, logo) semantics.

4. Probabilistic Program Induction

When building sites, skilled designers often rely on formalized knowledge about design patterns, typically encapsulated in books or style guides (van Duyne et al., 2002). Such rules for good design, however, are difficult to enumerate and operationalize. A more attractive proposition is to learn these rules from data by inducing a generative probabilistic model of pages. Recent work on concept learning and Bayesian program induction provides a promising avenue for learning design patterns from structured representations (Hwang et al., 2011).

Inducing a formal language of page designs from a corpus of exemplars would allow many complex design tasks to be formulated as probabilistic inference problems and solved using standard Monte Carlo techniques. For instance, a designer could construct a partial specification of a Web page, and a design tool could "autocomplete" the page by conditioning the model on the specification and performing MAP estimation.

5. Find Out More

We believe that data-driven Web design is an area ripe for future research. We thank our collaborators; find out more about our work at `http://bricolage.stanford.edu`.

References

Buxton, Bill. *Sketching User Experiences.* Morgan Kaufmann, 2007.

Hwang, Irvin, Stuhlmüller, Andreas, and Goodman, Noah D. Inducing probabilistic programs by Bayesian program merging. *CoRR*, abs/1110.5667, 2011.

Kumar, Ranjitha, Talton, Jerry O., Ahmad, Salman, and Klemmer, Scott R. Bricolage: Example-based retargeting for web design. In *Proc. CHI*. ACM, 2011a.

Kumar, Ranjitha, Talton, Jerry O., Ahmad, Salman, Roughgarden, Tim, and Klemmer, Scott R. Flexible tree matching. In *Proc. IJCAI*, 2011b.

Socher, Richard, Lin, Cliff C., Ng, Andrew Y., and Manning, Christopher D. Parsing natural scenes and natural language with recursive neural networks. In *Proc. ICML*, 2011.

van Duyne, Douglas K., Landay, James A., and Hong, Jason I. *The Design of Sites.* Addison-Wesley, 2002.