Web Intelligence Linked Open Data for Website Design Reuse

Maxim Bakaev[1(✉)], Vladimir Khvorostov[1], Sebastian Heil[2], and Martin Gaedke[2]

[1] Novosibirsk State Technical University, Novosibirsk, Russia
{bakaev, xvorostov}@corp.nstu.ru
[2] Technische Universität Chemnitz, Chemnitz, Germany
{sebastian.heil, martin.gaedke}@informatik.tu-chemnitz.de

Abstract. Code and design reuse are as old as software engineering industry itself, but it's also always a new trend, as more and more software products and websites are being created. Domain-specific design reuse on the web has especially high potential, saving work effort for thousands of developers and encouraging better interaction quality for millions of Internet users. In our paper we perform pilot feature engineering for finding similar solutions (website designs) within Domain, Task, and User UI models supplemented by Quality aspects. To obtain the feature values, we propose extraction of website-relevant data from online global services (DMOZ, Alexa, SimilarWeb, etc.) considered as linked open data sources, using specially developed web intelligence data miner. The preliminary investigation with 21 websites and 82 human annotators showed reasonable accuracy of the data sources and suggests potential feasibility of the approach.

Keywords: Linked data quality · Software reuse · Web design patterns · Data mining · Model-driven development

1 Introduction

Nowadays, web engineering (WE) has become an established, multi-billion dollar industry, and the number of websites worldwide exceeded 1 billion, although only a quarter of them are believed to be truly active. Given the multitude of existing websites and the amount of work effort put into producing and debugging the respective code up to date, their reuse would seem to be an attractive opportunity. Reuse is consistently named by Software Engineering (SE) experts among the advances and techniques that increased programmers' productivity the most, but its applicability on a large scale is domain-dependent [1]. On the web, the current stage of the industry development implies "mass-production" of functionality (code) and design, while content and usability need to be "hand-crafted" and their reuse seems problematic. Simple reuse of code is enabled via development environments and content management systems/frameworks, while more advanced methods involve self-organizing component-based WE (e.g. [2]), evolutionary programming, etc. Reuse of design, which is considered to

J. Cabot et al. (Eds.): ICWE 2017, LNCS 10360, pp. 370–377, 2017.
DOI: 10.1007/978-3-319-60131-1_22

be even more promising than reuse of code [1] and that is in our focus in the current research work, started to attract special interest in the 1990s and at the time was popularly shaped as design guidelines or patterns.

Recently, after data mining and content mining, the term *design mining* came to denote automated extraction of design patterns and trends from large collections of design examples. In case of the potent *Webzeigeist* tool that implements a kind of *design search engine*, designs are structured form web pages that are conveniently collected from the WWW, and then direct, query-based or stream-based access to design elements can be performed effectively [3]. The Webzeigeist authors rightfully claim that "in a database of ten million pages, the likelihood (that a designer will find a useful example) increases", but one shouldn't fall into the same pit as the early Internet search engines that valued results' quantity over relevance. Populating the database with web designs or design patterns shouldn't be a problem, given the currently existing billion of websites, but selecting the ones appropriate to the project context doesn't seem to be resolved.

In design example repositories, like Webzeigeist, search can be carried out on rather technical design parameters, like page aspect ratio or element styles. Extensive libraries of website templates, which have been named the "killers" of web design for more than a decade now, encompass many advanced tools (e.g. [4]), but suffer from the same organizational issues, as virtually none of them can adequately perform search based on problem description or design context. Thus, feature engineering is generally not performed in such collections, and moreover, there seems to be no agreed set of features for website reuse. In addition, there's a problem with identifying values for these features, especially for a website you don't own – in this case project specifications and website use logs are not available for data mining.

So, our current paper is a study in progress dedicated to identifying a set of features important for reuse of website design and finding the ways to obtain their concrete values. Particularly, we explore the feasibility of "web intelligence" (WI) approach, where mining of the website code is supplemented with extraction of website-related data (we rather not call them "metadata", since it denotes a different thing in HTML) from external sources. In Sect. 2, we overview feature engineering process for websites and propose model-based UI development approach as the appropriate framework. Then, we describe the architecture and capabilities of the dedicated web intelligence linked open data miner that we developed. In Sect. 3, we test the formulated hypotheses on some WI data to make inferences regarding the data accuracy and choosing data sources of higher quality. Finally, we make the conclusions and outline directions for further research work in the field.

2 Method

2.1 Feature Engineering for Website Design Reuse

There's a general consensus that feature engineering (FE) is crucial in applied machine learning, building recommender systems, case-based reasoning, etc. [5]. The major stages of the conventional FE process can be identified as: forming the excessive list of

potential features (e.g. through brainstorming session), implementing all or some of them in a prototype, and selecting relevant features by optimizing the considered subset. Then, the corresponding similarity (distance) calculation approaches may be used to retain, usually via AI methods, the website designs that are most relevant for the current web project and offer the best chance of reuse.

A fair amount of research works deal with feature selection for web pages, particularly for automated classification purposes [6, 7]. Indeed a web page is a technically opportune object for analysis, as it is represented in easily processable code (HTML, CSS, etc.), but it's not self-contained, either goal-wise or in terms of design resolutions. So, we believe that FE for reuse should be performed for a whole website (web project) and that model-based (MB) approach to web UI development provides a good starting point for assembling the potential features (since for a conventional website user, web interface basically equals website). The MB paradigm identifies three groups of models: (1) per se interface models – Abstract UI, Concrete UI, and Final UI, (2) functionality-oriented models – Tasks and Domain, and (3) context of use models – User, Platform, and Environment. Of these, we consider the Domain, Tasks, and User of higher relevance to website design reuse and will apply them in the FE, while Platform and Environment models rather relate to website's back-office. Also, not all existing website designs are equally good (in contrast to e.g. re-usable programming code), so quality aspects must be reflected in the feature set. Let us further consider the selection of features and the corresponding similarity calculation approaches in more detail.

Domain: theoretically, the domain of a website may be inferred from its content, but this is quite complex and computationally expensive problem. Alternatively, website classifications in major web catalogues may be used, with the distance (similarity measure) defined as the minimal number of steps to get from one category item to another via hierarchical relations, which can be then divided by the "depth" of the item, to reduce potential bias for less specifically classified websites.

Tasks: extraction of tasks from website code is probably the best developed one in reverse WE (e.g. [8]), and the resulting model is generally specified in UML. In the simplest case, given that the domain is known, conventional tasks can be represented with the website chapter labels extracted from the code and arranged as tag cloud, with the subsequent employment of well-developed semantic similarity/distance methods [9].

User: stereotype modeling (with FOAF, WebML, etc.) implies identifying user groups and developing the corresponding user profiles or "personas", where features commonly include gender, age, experience, education level, etc. The methods for assessing similarity between profiles of website or social network users are reasonably well developed [10], but concrete demographics of the target users for someone's website aren't easy to obtain. It's naturally available in web project's specifications, while real user behavior patterns can be mined from access/interaction logs [11], but without access to either, a popular approach is employment of human annotators.

Quality: for the purposes of reuse, two dimensions of quality may be identified: (1) website's intrinsic quality of implementation – how well it was made from technical perspective and (2) quality-in-use – how well the website performs in online

environment, satisfying target users in their tasks. The features for the former are reasonably well developed and quantifiable: website code correctness, accessibility, size of web pages, response times, etc. The latter closely matches the notion of usability whose concrete value is hard to auto-assess, but which is reflected in visitor behavior factors collected by web analytics services: bounce rates, average page views, conversion and completion rates, etc., though these data are generally not made openly available.

2.2 Linked Data Sources and Web Intelligence

Already more than a decade ago, the concept of publishing data in a semantic-aware, machine-readable form, ready for use by remotely accessing software, was shaped as Linked Data, and nowadays many web services, mashups, etc. rely on freely obtainable Linked Open Data (LOD). Finding an appropriate LOD source and estimating its quality is highly important in such web projects, but there's lack of research on the topic [12]. Some specific dimensions of LOD quality are: *Amount of Data*, *Conciseness*, *Completeness*, *Navigability* and *Interlinking* [12], but undoubtedly fitness for use and data accuracy are of foremost consideration for data users [13, 14].

As we mentioned before, the values for many of the features potentially significant for web design reuse are hard to determine in the absence of the website specifications and use statistics. However, virtually any operational website is regularly explored by crawlers, robots, spiders, etc. of numerous global web services, and is presented in web catalogues and search/indexing systems. For example, DMOZ catalogue claims to contain more than 1 million hierarchically-organized categories, and the number of included websites is about 4 million, which implies reasonably detailed classification – far more thorough than most website content analysis approaches could provide. Further, global web "aggregating" services, such as *Alexa* or *SimilarWeb*, are capable of indirectly estimating certain quality-related parameters even for websites with closed web statistics. Since the "fingerprints" of most websites are all around the web, the term *Web Intelligence* may be loosely applied to the process of website-related data gathering from LOD sources and their accuracy cross-checking.

To automate data collection, we developed a prototype WI miner capable of extracting data from specified locations, structuring them, and keeping them in the database. The current version (at http://webmining.khvorostov.ru) receives a website URL as the input, collects and structures the data (presented in Table 1), then outputs and stores them in the database. The main classes of the prototype, which in general correspond to the model-view-controller (MVC) pattern, are:

- AbstractController – abstract class for application controllers;
- SiteController – controller that displays the results;
- SiteAjaxController – controller responsible for processing AJAX queries;
- DBData – the model component class that interacts with the DB;
- IMiner – interface for implementation by all the classes related to mining (AlexaMiner, SimilarMiner, SectionsMiner, DMOZMiner, etc.);
- MinerFabric – factory class that returns the object of the necessary class for mining;
- AbstractHtmlParserMiner – abstract class for the miners that parse HTML.

Table 1. Web Intelligence data collected by the WI miner

Model/Realm	Features	WI sources
Domain	1. website category	DMOZ, SimilarWeb
Tasks	1. website chapter names, 2. number of website chapters	The website code (main navigation only)
User	1. demographics (Male, Female, No College, Some College, Graduate School, College), 2. Flesch-Kincaid Grade Level (~age)	Alexa (1), readability-score.com (2 – homepage only)
Quality	1. number of errors and warnings, 2. page load time, 3. bounce rate, 4. popularity rank (global), 5. number of visits	validator.w3.org (1 – homepage only), Alexa (2, 3, 4), SimilarWeb (3, 5)

2.3 The WI LOD Accuracy Investigation

The goal of our investigation was to perform preliminary analysis of the LOD sources accuracy, by testing them against some "common sense" from the WE field. To this end, we decided to supplement the data collected by the WI miner with website usability evaluations provided by human annotators, considered to be representative of the quality-in-use. Specifically, we employed official websites of 11 German and 10 Russian universities (all English versions) and 82 annotators representing the target user group (more detailed description of the experimental setup can be found in [15]). The reason we decided not to vary the Domain was that accuracy of website category data is obvious (the validity of the similarity measure based on these data is a different issue). So, the following hypotheses (H_i) were formulated for the WI LOD:

Cross-checking: analogous values provided by Alexa and SimilarWeb (the two *bounce rates* and *popularity rank* vs. *number of visits*) should correspond to each other (H_1).

Domain: none – the factor was fixed as *Career and Education* (SimilarWeb).

Tasks: more straightforward *website chapter names* should result in lower *bounce rates* (H_2) and higher *usability evaluation* (H_3).

User: since web content is important in user subjective impression of a website, *Flesch-Kincaid Grade Level* should affect *usability evaluation* (H_4). Given the target audience of university websites and presumably their dedicated usability engineering, higher share of *College*-level users should result in higher *usability evaluation* (H_5).

Quality: the technical quality (*number of errors and warnings*, *page load time*) and the quality-in-use factors (*bounce rates*, *popularity rank*, *number of visits*, *usability evaluation*) should be positively correlated (H_6).

3 Results

3.1 The Data Validity and Cross-Checking

Our preliminary analysis of the data validity found one outlier – a website for which the extracted *number of visits* was at 28 visitors per month and *bounce rate* (SimilarWeb) was at 100%. So, the 20 websites (95.2% of the data) were valid for the analysis. Also, user education-related data extracted from Alexa was incomplete, as only *College* and *Graduate School* were available for all the websites.

H_1: correlation between *bounce rate* values extracted from Alexa and from SimilarWeb was $r = 0.582$ ($p = 0.007$), while correlation between the *popularity rank* and the *number of visits* was $r = -0.600$ ($p = 0.005$).

3.2 The Tasks Model

To pinpoint the tasks that correspond to the *Career and Education* domain (SimilarWeb), we identified 8 most typical chapter labels, which were found on 6 or more of the websites: *University/About us* (present on 21 websites), *Research/Science* (19), *International* * (12), *Study* (10), *Faculties* (8), *Prospective students/Admissions* (7), *Contacts* (6), *News/Media/Press* (6). Then for each website we divided the number of the typical chapters it has by the total number of chapter in its main navigation, thus receiving the website "conventionality" value (ranging from 0.4 to 1, average 0.622, SD = 0.159).

H_2: negative correlation between the website "conventionality" and the SimilarWeb *bounce rate* was significant ($p = 0.002$, $r = -0.587$), unlike for Alexa *bounce rate*. Also, we found no significant correlation with the *usability evaluation* (H_3).

3.3 The User Model

H_4: correlation between the *Flesch-Kincaid Grade Level* and the *usability evaluations* was significant ($p = 0.01$; $r = 0.561$), which may imply positive effect of sophisticated texts on website evaluation by the target group.

H_5: correlation between the *College* share and the *usability evaluations* was significant at $\alpha = 0.06$ ($p = 0.056$; $r = 0.433$),

3.4 Quality

H_6: somehow unexpectedly, we found significant negative correlation between the *number of errors and warnings* (summarized) and the *bounce rate* extracted from Alexa ($p = 0.033$; $r = -0.479$). Also, correlation between *popularity rank* and *page load time* was significant at $\alpha = 0.08$ ($p = 0.078$; $r = 0.403$).

Further exploring whether usability evaluations provided by annotators can be predicted by the mined WI LOD, we constructed the regression model with comprehensive list of factors, using the *Backwards* inclusion method. We selected the model

that had the highest adjusted R^2, and it included 4 factors: the *Flesch-Kincaid Grade Level* (FK), the Alexa *College* share (C), the *number of errors and warnings* (E), and the *number of visits* (V, in millions). The model was significant (p = 0.01), but had moderate $R^2 = 0.559$:

$$U_{eval} = 2.94 + 0.09 * FK + 0.79 * C - 0.01 * E - 0.19 * V \quad (1)$$

4 Discussion and Conclusions

The general idea of design reuse seems to be repeatedly re-invented at different stages of SE industry development under different names and in various sub-fields. Currently, existing website design repositories focus on technical, structural or stylistic aspects, but not on problem- or user-oriented ones; neither they contain a quality metric. Since code analysis alone can't provide values for most features important for retaining appropriate solutions for reuse, and employment of human annotators is restricted in scale and budget, our proposal is to gather website-related data (metadata, in non-technical sense) from linked open data sources. In this, our first step in the current pilot research work was to investigate the accuracy of the data sources and general feasibility of the approach.

To this end, we developed prototype "web intelligence" data miner capable of extracting data (see Table 1) for any given website from Alexa, SimilarWeb and certain other online services. To test the accuracy of the data, we performed automated data collection for 21 German and Russian university websites (*Career and Education* domain) and asked 82 human annotators who represented a target user category – students – to provide subjective evaluations of the websites' usability. More information on the experimental setup can be found in [15], while the extracted data and some supplementary materials are available at http://webmining.khvorostov.ru/docs.zip. In regard to the 6 hypotheses formulated for accurate data, the results of the analysis showed the following:

H_1 (cross-checking): effect found, at reasonably high statistical significance.

H_2 and H_3 (Tasks): effect found only for *bounce rate* provided by SimilarWeb (in line with usability guidelines about not making users think any more than necessary), which should be considered the preferred data source.

H_4 and H_5 (User): effects found (arguably appropriate for the target user group).

H_6 (Quality): few effects found, and the effect for Alexa was the opposite of what was expected. The regression model for the *usability evaluation* with the quality factors was significant (p = 0.01, $R^2 = 0.559$). *Usability evaluation* was negatively affected by the *number of errors and warnings* as well as by higher *number of visitors* (we can only speculate that websites of smaller universities may be better tailored to the needs of their target users).

So, we suggest that WI data mined from web analytic services, search engines, even advertisement networks, can indeed be a useful supplement to analysis of the actual website code and manual annotations, when website design similarity is evaluated for

the purposes of reuse. Limitations of the current pilot research include quite a small number of employed websites (since we were basically limited by the effort of human annotators) and rather informal approach to data accuracy analysis. In our further research we also plan to focus on auto-engineering features (deep learning) and finding out the values for the User model that seems to be under-explored in modern literature, for which we plan to extend the capabilities of our prototype WI data miner to gather data from web analytic services. Moreover, the analysis of the Domain model should include both similarity calculation in hierarchical web catalogues and, possibly, natural language processing of titles and descriptions submitted by website owners.

Acknowledgement. The reported study was funded by RFBR according to the research project No. 16-37-60060 mol_a_dk. The authors also thank S. Firmenich and J.M. Rivero from LIFIA (Argentina) who contributed to the discussion of the paper topics.

References

1. Glass, R.L.: Facts and Fallacies of Software Engineering. Addison-Wesley Professional, Boston (2002)
2. Gaedke, M., Rehse, J.: Supporting compositional reuse in component-based Web engineering. ACM Symp. Appl. Comput. **2**, 927–933 (2000)
3. Kumar, R., et al.: Webzeitgeist: design mining the web. In: SIGCHI Conference on Human Factors in Computing Systems, pp. 3083–3092 (2013)
4. Norrie, M.C., Nebeling, M., Geronimo, L., Murolo, A.: X-Themes: supporting design-by-example. In: International Conference on Web Engineering (ICWE 2014), pp. 480–489 (2014)
5. Anderson, M.R., et al.: Brainwash: a data system for feature engineering. In: 6th Biennial Conference on Innovative Data Systems Research (2013)
6. Mangai, J.A., Kumar, V.S., Balamurugan, S.A.: A novel feature selection framework for automatic web page classification. Int. J. Autom. Comput. **9**(4), 442–448 (2012)
7. Saraç, E., Özel, S.A.: An ant colony optimization based feature selection for web page classification. Sci. World J., **2014**, 1–16 (2014). doi:10.1155/2014/649260, Article ID: 649260
8. Paganelli, L., Paterno, F.: A tool for creating design models from web site code. Int. J. Softw. Eng. KEng. **13**(02), 169–189 (2003)
9. Park, J., Choi, B.C., Kim, K.: A vector space approach to tag cloud similarity ranking. Inf. Process. Lett. **110**(12), 489–496 (2010)
10. Kosinski, M., et al.: Manifestations of user personality in website choice and behaviour on online social networks. Mach. Learn. **95**(3), 357–380 (2014)
11. Varnagar, C.R., et al.: Web usage mining: a review on process, methods and techniques. In: IEEE Information Communication and Embedded Systems (ICICES), pp. 40–46 (2013)
12. Cappiello, C., Di Noia, T., Marcu, B.A., Matera, M.: A quality model for linked data exploration. In: International Conference on Web Engineering (ICWE), pp. 397–404 (2016)
13. Zaveri, A., et al.: Quality assessment for linked data: a survey. Seman. Web **7**(1), 63–93 (2016)
14. Wang, R.Y., Strong, D.M.: Beyond accuracy: what data quality means to data consumers. J. Manage. Inf. Syst. **12**(4), 5–33 (1996)
15. Bakaev, M., Gaedke, M., Heil, S.: Kansei Engineering experimental research with University websites. TU Chemnitz Technical Report, CSR-16-01 (2016)