Int. J. Human-Computer Studies 84 (2015) 51–66

Contents lists available at ScienceDirect

Int. J. Human-Computer Studies

journal homepage: www.elsevier.com/locate/ijhcs

How to apply Markov chains for modeling sequential edit patterns in collaborative ontology-engineering projects ☆, ☆☆

Simon Walk [a,*], Philipp Singer [b], Markus Strohmaier [b,c], Denis Helic [d], Natalya F. Noy [e], Mark A. Musen [e]

[a] Institute for Information Systems and Computer Media, Graz University of Technology, Austria
[b] GESIS – Leibniz Institute for the Social Sciences, Cologne, Germany
[c] Department of Computer Science, University of Koblenz-Landau, Germany
[d] Knowledge Technologies Institute, Graz University of Technology, Austria
[e] Stanford Center for Biomedical Informatics Research, Stanford University, USA

ARTICLE INFO

Article history:
Received 4 June 2014
Received in revised form
27 July 2015
Accepted 28 July 2015
Communicated by Scott Bateman
Available online 20 August 2015

Keywords:
Markov chains
Sequential patterns
Usage patterns
Collaborative ontology engineering

ABSTRACT

With the growing popularity of large-scale collaborative ontology-engineering projects, such as the creation of the 11th revision of the International Classification of Diseases, we need new methods and insights to help project- and community-managers to cope with the constantly growing complexity of such projects. In this paper, we present a novel application of Markov chains to model sequential usage patterns that can be found in the change-logs of collaborative ontology-engineering projects. We provide a detailed presentation of the analysis process, describing all the required steps that are necessary to apply and determine the best fitting Markov chain model. Amongst others, the model and results allow us to identify structural properties and regularities as well as predict future actions based on usage sequences. We are specifically interested in determining the appropriate Markov chain orders which postulate on how many previous actions future ones depend on. To demonstrate the practical usefulness of the extracted Markov chains we conduct sequential pattern analyses on a large-scale collaborative ontology-engineering dataset, the International Classification of Diseases in its 11th revision. To further expand on the usefulness of the presented analysis, we show that the collected sequential patterns provide potentially actionable information for user-interface designers, ontology-engineering tool developers and project-managers to monitor, coordinate and dynamically adapt to the natural development processes that occur when collaboratively engineering an ontology. We hope that presented work will spur a new line of ontology-development tools, evaluation-techniques and new insights, further taking the interactive nature of the collaborative ontology-engineering process into consideration.

1. Introduction

In recent years, we have seen significant increase in the use of structured data. In many cases, workers have used ontologies to integrate and interpret this data. As a result, we have seen an increase in the number of large-scale projects, focusing on collaboratively engineering ontologies. For example, the World Health Organization (WHO) is leading the collaborative online development of the new revision of the International Classification of Diseases (ICD), which represents an important classification scheme that is used in many countries around the world for health statistics, insurance billing, epidemiology, and so on. Wikidata,[1] another collaborative ontology-engineering project initiated by the Wikimedia Foundation,[2] is gathering structured data in multiple languages to link to and between Wikipedia and its different language editions. To understand and support the new requirements that this collaborative approach introduces, researchers have analyzed and developed new ontology-engineering tools, such as Protégé and WebProtégé (Tudorache et al., 2008, 2011). These tools not only provide a collaborative environment to engineer ontologies, but also include mechanisms that are targeted towards augmenting

☆This paper has been recommended for acceptance by Scott Bateman.
☆☆An earlier version of this paper was submitted to arXiv on 05.03.2014 and can be found at http://www.arxive.org/abs/1403.1070/.
* Corresponding author.
E-mail address: simon.walk@tugraz.at (S. Walk).

[1] http://www.wikidata.org
[2] http://wikimediafoundation.org

http://dx.doi.org/10.1016/j.ijhcs.2015.07.006
1071-5819/

collaboration and increasing the overall quality of the resulting ontologies by supporting contributors in reaching consensus. For user-interface designers, community managers as well as project administrators, analyzing and understanding the ongoing processes of how ontologies are engineered collaboratively is crucial. When provided with detailed and quantifiable insights, the used ontology-engineering tools or even the development strategy can be automatically revised and adjusted accordingly. Engineering an ontology by itself already represents a complex task; this task becomes even more complex when adding a layer of social interactions on top of the development process. In the light of these challenges, we need new methods and techniques to better understand and measure the social dynamics and processes of collaborative ontology-engineering efforts.

In this work, we want to focus on sequences of actions that users perform when collaboratively engineering ontologies. For example, when the change of a property by a user is succeeded by another change of a property by that user, the two changes can be used to represent the sequence of properties that this specific user has been working on. Better understanding such sequential processes can help system designers to increase the quality of an ontology or contributor satisfaction, among other things. To come back to our previous example, if we better understand the process of how users sequentially edit properties of concepts, we can recommend to users the property that they potentially might want to edit next. Alternatively, we can steer users away from their typical behavior in order to cover niche parts of the ontology. We know from previous studies, that sequential patterns of human actions can usually be predicted quite well. For example, Song et al. (2010) showed that human mobility patterns are predictable; they also hypothesize that all human activities contain certain regularities that can be detected. We explore whether these regularities might also apply to our ontology-editing sequences.

Consequently, our main goal in this paper is the presentation of methods and techniques for acquiring detailed insights into these ongoing (sequential) processes when users collaboratively engineer an ontology. Hence, we introduce a novel application of a methodology based on Markov chains. We base our elaboration of this method on previous work that has focused on studying human navigational paths through websites (Singer et al., 2014). We focus not only on the structure of given paths (e.g., the identification of common sequences), but also on the detection of memory (e.g., on how many previous changed properties does the next property a user changes depend on). We lay our focus on determining the appropriate Markov chain orders which allows us to get insights into on how many previous actions users reason their future actions. The *main objectives* of this paper are:

- The presentation of a novel application of Markov chains on the change logs of collaborative ontology-engineering projects to gather new insights into the processes that occur when users collaboratively create an ontology.
- The demonstration of the utility of the presented and adapted Markov chain framework by applying it on a large scale collaborative ontology-engineering project.

Tackling these two objectives enables us to answer questions that are of practical relevance for the development of collaborative ontology-engineering tools, such as Do users have to switch frequently between the user-interface sections when working on the ontology? Which concept is a user likely to change next, the one closer to or further away from the root concept of the ontology? Which change type is a user most likely to perform next? Do users move along the ontological hierarchy when changing content? Can we identify edit behaviors, such as *top-down* or *bottom-up* editing? Do users only reason their future actions on the current ones or do they depend on a series of preceding ones? However, other kinds of questions are conceivable and can be studied in straight-forward manner by researchers by focusing on the methodological aspects presented in this work.

Results: Our results indicate that the application of Markov chains on the change-logs of collaborative ontology-engineering projects provides new and potentially actionable insights into the processes that occur when users collaboratively create an ontology for project administrators and ontology-engineering tool developers.

Contributions: We provide (i) a detailed description of the process for applying Markov chains on the change-logs of collaborative ontology-engineering projects and (ii) an evaluation of the extracted Markov chain models by applying the methodology on the change-logs of ICD-11, representing a large-scale collaborative ontology-engineering project that exhibits Markov chains of varying orders. Our *high-level contribution* is the presentation of a novel approach that can be used to gather new insights into ongoing processes when collaboratively engineering an ontology by making use of Markov chains to model sequential usage sequences. Amongst others, this allows practitioners to identify structural properties and regularities as well as predict future actions based on usage sequences.

The remainder of the paper is structured as follows: In Section 2 we provide a brief introduction into collaborative ontology-engineering. We then continue to review related work in Section 3. In Section 4, we briefly describe and characterize the history of ICD-11 as well as the dataset and the underlying change-log. We continue with the description of the process in Section 5, describing all necessary steps to extract and interpret Markov chains for a given dataset. In Section 6, we apply the previously described process to ICD-11, extracting Markov chains of different orders for two different types of analyses. In Section 7, we discuss potential implications and conclude our work in Section 8.

2. Collaborative ontology engineering

According to Gruber (1993), Borst (1997) and Studer et al. (1998), an ontology is an explicit specification of a shared conceptualization. In particular, this definition refers to a machine-readable construct (the formalization) that represents an abstraction of the real world (the shared conceptualization), which is especially important in the field of computer science as it allows a computer (among other things) to "understand" relationships between entities and objects that are modeled in an ontology.

The field of collaborative ontology engineering and its environment pose a new field of research with many new problems, risks and challenges. In general, contributors of collaborative ontology-engineering projects, similar to other collaborative online production systems (e.g., Wikipedia), engage remotely (e.g., via the internet or a client–server architecture) in the development process to create and maintain an ontology. Given the complexity assigned to engineering an ontology, researchers and practitioners have already discussed and proposed different development methodologies. Analogously to the plethora of different software development processes and methodologies (i.e., the Waterfall-Model, agile development or SCRUM), methodologies and guidelines exist for (collaboratively) creating an ontology which define multiple different aspects of the engineering process. For example, the *Human-centered ontology engineering methodology* (HCOME) (Kotis et al., 2005; Kotis and Vouros, 2006; Kotis, 2008) represents such an approach that sets its focus on (continuously and) actively integrating the knowledge worker—the users who will rely on and use the created ontology—in the ontology life-cycle (i.e., by including the users in all planning stages, discussions, requirements analyses, etc.). Similarly, the *DILIGENT* process (Pinto et al.,