Analysis of User Editing Patterns in Ontology Development Projects

Hao Wang[1], Tania Tudorache[2], Dejing Dou[1],
Natalya F. Noy[2], and Mark A. Musen[2]

[1] Department of Computer and Information Science
1202 University of Oregon, Eugene, OR 97403, USA
{csehao,dou}@cs.uoregon.edu
[2] Stanford Center for Biomedical Information Research
Stanford University, 1265 Welch Road, Stanford, CA 94305, USA
{tudorache,nyulas,musen}@stanford.edu

Abstract. The development of real-world ontologies is a complex undertaking, commonly involving a group of domain experts with different expertise that work together in a collaborative setting. These ontologies are usually large scale and have a complex structure. To assist in the authoring process, ontology tools are key at making the editing process as streamlined as possible. Being able to predict confidently what the users are likely to do next as they edit an ontology will enable us to focus and structure the user interface accordingly and to enable more efficient interaction and information discovery. In this paper, we use data mining techniques to investigate whether we are able to predict the next editing operation that a user will make based on the change history. We have analyzed the change logs of two real-world biomedical ontologies, and used association rule mining to find editing patterns using different features. We evaluated the prediction accuracy on a test set of change logs for these two ontologies. Our results indicate that we can indeed predict the next editing operation a user is likely to make. We will use the discovered editing patterns to develop a recommendation module for our editing tools, and to design user interface components that are better fitted with the user editing behaviors.

1 Collaborative Ontology Development and Related Work

Distributed and collaborative development by teams of scientists is steadily becoming a norm rather than an exception for large ontology-development projects. In domains such as biomedicine the majority of large ontologies are authored by groups of domain specialists and knowledge engineers. The development of ontologies such as the Gene Ontology (GO) [7], the National Cancer Institute Thesaurus (NCI Thesaurus) [17] and the International Classification of Diseases (ICD-11) deploys varying collaborative workflows [16]. Many of these projects have several things in common: First the ontologies are very large (e.g., GO

R. Meersman et al. (Eds.): OTM 2013, LNCS 8185, pp. 470–487, 2013.

has over 39,000 classes; ICD-11 has over 45,000). Second, many users who contribute to the ontologies are not themselves ontology experts and they do not see ontology development as part of their day-to-day jobs. Indeed, the majority of contributors to ICD-11, for example, are medical professionals. At the same time, researchers have long contended that ontology development is a cognitively complex and error prone process [6,15]. The overarching goal of our research on collaborative ontology development is to develop methods that facilitate this process and make it more efficient for users.

In this paper, we explore the use of structured change logs to predict the changes that users are likely to make next. Ontology change logs provide an extremely rich source of information. We and other investigators have used change data from ontologies to measure the level of community activity in biomedical ontologies [11], to migrate data from an old version of an ontology to a new one [10], and to analyze user roles in the process of collaboration [4,5,18,21]. For example, we have demonstrated that we can use the change data to assess the level of stabilization in ontology content [21], to find implicit user roles [5], and to describe the collaboration qualitatively [18]. For example, we found that changes to ICD-11 tend to propagate along the class hierarchy: A user who alters a property value for a class is significantly more likely to make a change to a property value for a subclass of that class than to make an edit anywhere else in the ontology [14]. Similarly, Pesquita and Couto found that structural features and the locations of changes in the Gene Ontology are predictive of where future changes will occur [13]. Cosley and colleagues developed an application that provided specific suggestions to Wikipedia editors regarding new articles to which they might want to contribute [3]. The model aggregated information about the users, such as preferences and edit history. The researchers found that recommendations based on models of the user editing behaviors made the contributors four times more likely to edit any article compared with random suggestions.

We explore the following hypothesis in this paper: In large collaborative ontology development projects, there are patterns of change that persist over time, across different users, and across different projects/ontologies. We evaluate a set of features for ontology changes, such as properties being edited, and evaluate their predictive power. We use association rule mining, a popular data mining technique to extract the patterns based on these features. Association rules provide straightforward guidance to the user-interface designer by suggesting editing patterns. Indeed, we focus on the features and the types of pragmatic patterns that help us build more efficient interface for ontology development.

Specifically, this paper makes the following contributions:

- We develop a data mining method to predict change patterns in collaborative ontology development (Section 3).
- We propose a set of features for association rules that describe change patterns in collaborative ontology development (Section 3).
- We evaluate our method by analyzing a large number of changes from change logs on two large real-world ontology development projects that are run by the World Health Organization (WHO) (Section 4).

2 Preliminaries

We start by providing background on iCAT, which is a custom-tailored version of WebProtégé [20], a tool that we designed for collaborative ontology development. Today hundreds, if not thousands of users, rely on WebProtégé in their projects (Section 2.1). We then describe the two large ontologies that use WebProtégé and that we used in our evaluation. These are two ontologies in the Family of International Classifications that are developed and maintained by the WHO (Section 2.2). Finally, we provide background on association rule mining (Section 2.3), a technique that we use to find patterns of changes.

2.1 WebProtégé

In this paper, we analyze the data from two ontologies that are developed using a custom-tailored version of WebProtégé [20], which is itself a web-based version of Protégé, the most widely used open-source ontology-editing environment. WebProtégé enables users to edit ontologies in their web browser in a distributed fashion. Users can contribute to the ontology simultaneously, comment on each other's edits, maintain discussions, monitor changes and so on. One of the key features of WebProtégé is the ability of project administrators to custom tailor the user interface to suit the needs of a particular project. Specifically, in this paper we focus on the two ontologies that are developed in iCAT, a version of WebProtégé that is custom tailored to the data model that the WHO uses. Figure 1 shows a screenshot of a panel for editing classes in iCAT. Because each class (e.g., disease description) has as many as 56 properties defined in the data model, iCAT groups these properties visually into "tabs" in the user interface. Each tab is used for editing values for properties in the same *property category*. For example, the `Title & Definition` tab in Figure 1 shows the properties in the category with the same name: `ICD-10 Code`, `Sorting label`, `ICD Title`, `Short Definition` and `Detailed Definition`. The `Clinical Description` tab and property category contains the properties: `Body system`, `Body part` and `Morphology`. iCAT has 15 such tabs and corresponding property categories.

Protégé (and, hence, iCAT) keeps a detailed structured log of changes and their metadata [12] shown in Figure 2. This log contains information about the content of the change and its provenance. We focus on changes to property values in the editing of ICD-11 and ICTM, by far the most frequent operation performed by the users. For example, in ICD-11 from 182,835 total changes, 180,896 are property changes. An example of a property value change tracked by iCAT is shown in the first row of Figure 2: *Replaced Sorting label of DB Acute myocardial infarction. Old value: DB. New value: BB.* For each property-value change, Protégé records the following structure information: *property name*, *class identifier* where the change occurred, the *old* and *new value*, the *author*, and *timestamp* of the change. Based on the user interface configuration (which follows the underlying data model), there is a unique association between a property

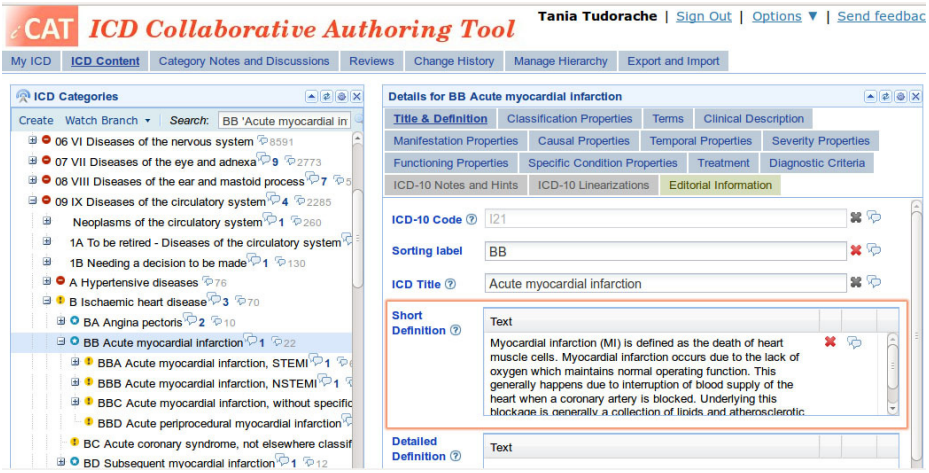


Fig. 1. The iCAT user interface used for editing the ICD-11 and ICTM ontologies. The left panel shows the disease class hierarchy and the right panel shows the properties of the selected disease class. The properties are organized in different tabs. Each tab (e.g., `Title & Definition`) corresponds to a property category with the same name. A property category contains several properties that are displayed in the respective tab. For example, the `Title & Definition` category contains the properties `ICD-10 Code`, `Sorting label`, `ICD Title`, `Short Definition` and `Detailed Definition`.

and a property category, so we can easily associate to each change the property category in which it occurred.

However, Protégé is not a requirement for the method that we describe in this paper; it is the presence of a detailed log of changes that is a requirement for the type of data mining that we present. As long as an ontology has a detailed structured log of changes available—regardless of the development environment that its authors use—it is amenable to association rule mining that we describe.

2.2 Ontologies: ICD-11 and ICTM

The 11th Revision of the International Classification of Disease (ICD-11) [1], developed by the World Health Organization, is the international standard for diagnostic classification that health officials in all United Nations member countries use to encode information relevant to epidemiology, health management, and clinical use. Health officials use ICD to compile basic health statistics, to monitor health-related spending, and to inform policy makers. As a result, ICD is an essential resource for health care all over the world. ICD traces its origins to the 19th century and has since been revised at regular intervals. The current in-use version, ICD-10, the 10th revision of the ICD, contains more than 20,000

[1] `http://www.who.int/classifications/icd/ICDRevision/`

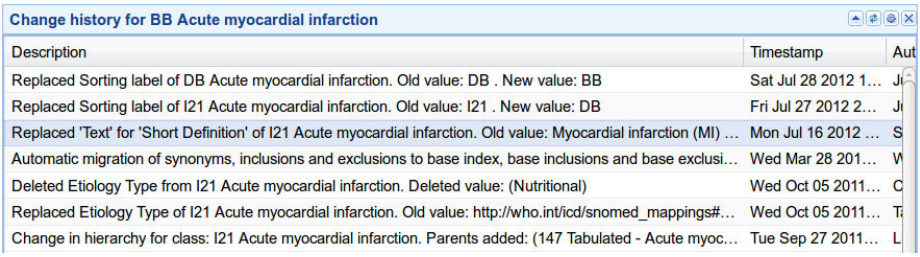


Fig. 2. Protégé (and iCAT) track every change in the system in a structured format. A change record has a textual description, a timestamp and an author, as well as other metadata not shown in this screenshot.

terms. The development of ICD-11 represents a major change in the revision process. Previous versions were developed by relatively small groups of experts in face-to-face meetings. ICD-11 is being developed via a web-based process with many experts contributing to, improve, and reviewing the content online [19]. It is also the first version to use OWL as its representation format.

The International Classification of Traditional Medicine (ICTM) is another terminology in the WHO Family of International Classifications. Its structure and development process is very similar to that of ICD-11. However, it is a smaller project, which was started later than the ICD-11 project. Thus, it has benefitted from the experiences of ICD-11 development and it used the tools that were already built for ICD-11. ICTM will provide an international standard terminology as well as a classification system for Traditional Medicine that can be used for encoding information in health records and as a standard for scientific comparability and communication, similar to ICD-11. Teams of domain experts from China, Japan and Korea are collaborating on a web platform with the goal of unifying the knowledge of their own traditional medicines into a coherent international classification. Even though ICTM shares some of the structures with ICD-11, there are many characteristics that are specific only for traditional medicine. ICTM is also developed concurrently in four different languages (English, Chinese, Japanese and Korean).

Data sources. We used the change logs generated by iCAT for both ICD-11 and ICTM. Table 1 shows some statistics about the ontologies themselves and their change logs. As the statistics show, ICTM is a smaller project compared to ICD-11. While ICTM has around 1,500 classes, ICD-11 has over 45,000. ICD-11 has also a deeper class hierarchy with 11 levels, compared to ICTM which has 7 levels. ICTM had a small number of users (12) who were making actively changes for the period of our data, while ICD-11 had 90 such users. The number of property changes which we use for our analysis also differ a lot: ICTM has 21,466 changes, while ICD-11 has 180,896.

Table 1. Ontology and change history statistics for ICTM and ICD-11

Data source characteristic	ICTM	ICD-11
Number of classes	1,511	45,028
Depth of ontology (number of levels)	7	11
Number of users	12	90
Time period	2/7/2011 - 8/21/2011	11/19/2009 - 5/24/2012
Total number of changes	26,607	182,835
Total number of property edit changes	21,466	180,896

2.3 Association Rule Mining

The change logs generated by iCAT provide a wealth of information that we can use to extract change patterns. These patterns of change can enable us to predict what operation the user is likely to perform next, based on her current operation and other features. We used *data mining* for the pattern-discovery task. Data mining is "the process of discovering interesting patterns from a large database" [9].

Association Rule Mining is a data mining technique that explores frequent patterns in large transactional data. The frequent patterns are usually expressed in terms of the combinations of features with certain values that appear together more frequently than the others. Agrawal and his colleagues introduced association rule mining in 1993 [1] and developed the Apriori Algorithm, a fast association rule mining algorithm, in 1994 [2]. The rules were presented in the form of inference rules with quantitative values to indicate the measure of interestingness. In the past decades, researchers have shown that association rules can discover and predict patterns with high efficiency and accuracy [9].

Let D be the set of n data tuples $D = t_1, t_2, ..., t_n$, where $t_i \subseteq \mathcal{I}$. $\mathcal{I} = i_1, i_2, ..., i_m$ is the set of features we want to discover the associations on. Let X and Y be two disjoint events such that $X \subset \mathcal{I}, Y \subset \mathcal{I}$ and $X \cap Y = \emptyset$. An association rule is an implication, $X \Rightarrow Y$, where X is called the antecedent and Y is called the consequent. The antecedent and consequent are conjunction of conditions on disjoint events. The rule provides the information on how likely Y is, given that we observed X. For example, if a user edits the *title* for a class (X), she may be likely to edit its *definition* next (Y). Therefore, association rule mining is a promising approach to predict the next editing operation that a user will make given the previous change logs.

It is common to use qualitative measures of interestingness in order to rank and filter association rules. Two of the most popular measures of interestingness are support and confidence. The **support** of an association rule $supp(X \Rightarrow Y)$ is a measure of how frequently the set of involved items appears in the data. Given event set X, support $s(X)$ is defined as the fraction of tuples $Ti \in D$ such that $X \subset Ti$. For rule $X \Rightarrow Y$,

$$support(X \Rightarrow Y) = P(X \cap Y)$$

is defined as a percentage of data tuples $X \cap Y$; in other words, it is the probability that both X and Y happens. Support is used to filter out association rules with too few occurrences because these rules do not provide enough information about the data and they are usually rare patterns.

Confidence is a measure of how precise these rules are. For rule $X \Rightarrow Y$,

$$Confidence(X \Rightarrow Y) = P(Y|X) = P(X \cap Y)/P(X)$$

In other words, confidence is the probability of Y given that X happens.

3 Method Description

The main goal of our analysis is to predict what the user is likely to do next given her current action. Therefore, our data tuples are *transitions* from one action to the next. Each transition in our set captures two operations from the structured change log: the features describing the current operation that the user performed and the features describing the next operation. We look for co-occurrences of features of the current operation and the next operation. For example, if the user edited the *title* of a class and then edited the *definition*, then the first edit is the current operation and the edit of the definition is the next one.

3.1 Data Preprocessing

We start our data processing by performing the following two preprocessing steps: (1) *feature extraction* and (2) *data aggregation*. The first step extracts the prediction related features from change log entries. The second step aggregates goal-irrelevant data into one data entry which will result in a cleaner and more goal-concentrated result.

Feature Extraction. A typical entry for a property change in the ICD-11 ontology (Figure 2) contains: (1) the information on the user who performed the change, (2) the timestamp, (3) the class identifier on which the change occurred, and (4) a textual description of the change. The latter item, the key source of features for our analysis, looks as follows:

Replaced ***'Text'*** *for* ***'Short Definition'*** *of* ***I21 Acute myocardial infarction***. ***Old value****: Myocardial infarction (MI) is defined as of heart muscle cells. Myocardial infarction occurs ...* ***New value****: Myocardial infarction (MI) is defined as the death of heart muscle cells. Myocardial infarction occurs...*

To use this log entry in our data mining analysis, we need the structured information that the change log provides and the additional features that we extract from the change description text. For example, for the change entry from the example, we extract the property on which the change occurred (`Short Definition`)

and we associate to it the property category (`Title & Definition`). We then analyze the next change performed by the same user—represented by a similar string—to extract the same feature about the next operation, as well as the feature reflecting whether the next change occurred in the same class or a different one.

As a result, we generate five features (see Table 2). Two features describe the current change—the *antecedent features*—and three features that describe the next change and the transition information—the *consequent features*.

Table 2. The 5 extracted features for each record in the change log that are used for association mining

Feature	Description of feature
`NAME_OF_PROPERTY` (antecedent)	The name of the edited property (Example: `Short Definition`)
`CATEGORY_OF_PROPERTY` (antecedent)	The category of the edited property (Example: `Title & Definition`)
`NEXT_NAME_OF_PROPERTY` (consequent)	The name of the next edited property (Example: `Body System`)
`NEXT_CATEGORY_OF_PROPERTY` (consequent)	The category of the next edited property (Example: `Clinical Description`)
`NEXT_ENTITY` (consequent)	A boolean flag that describes if the next edit operation is on the same entity as the previous change, or not. (Possible values: `Same` or `Not the same`)

Data Aggregation. The data change log provides abundant information that captures all aspects of user editing behaviors. For example, the user might edit a few characters of a property value, click elsewhere, and then come back and continue editing the same property. This behavior will result in two log entries describing consecutive edits to the same property. In reality though, it is usually just one editing operation from the user's point of view. We define a *consecutive operation* as two editing operations by one user on the same entity and the same property or category of property within a certain time interval (e.g., one hour). We aggregate such consecutive operations into a single operation.

Datasets for Rule Mining. The aggregated data with selected five features are ready for association rule mining. In our work, the data processing step generates four independent data sets. For each ontology (i.e., ICTM or ICD-11), we generated two datasets: one dataset with the operations aggregated based on property category and another one aggregated on property name.

3.2 Association Rule Mining: Apriori Algorithm

We generate the association rules by using the Apriori algorithm [2]. We use WEKA [8], the open source data mining software to generate association rules.

The Apriori algorithm contains two steps: *find all frequent itemsets* and *generate strong association rules from frequent itemsets*. An item is defined as a feature with assignment values, such as `CATEGORY_OF_PROPERTY= Temporal Properties`. An itemset is the conjunction of items. The Apriori Algorithm take two threshold as input: *t_support* and *t_confidence*. The *find all frequent itemsets* step will generated all the possible itemset I that satisfy support$(I) > t_support$. It uses the downward closure property of frequent itemsets: itemsets with more features are generated from frequent itemsets with fewer features. This property greatly reduces the search space and lowers the algorithm complexity. After finding all the frequent itemsets, the *find strong association rules* step divides the features in the frequent itemset I into two disjoint sets: antecedent X and consequent Y. We test the condition $confidence(X \Longrightarrow Y) > t\_confidence$ to generate the association rules.

The following is an example of an association rule generated by WEKA based on ICTM data:

$$\texttt{CATEGORY\_OF\_PROPERTY} = \texttt{Temporal Properties}\ 101 \implies$$
$$\texttt{NEXT\_CATEGORY\_OF\_PROPERTY} = \texttt{Diagnostic Method}\ \texttt{NEXT\_ENTITY} = \texttt{same}\ 70$$
$$conf : (0.69)$$

This rule indicates an association between the feature `CATEGORY_OF_PROPERTY`, `NEXT_CATEGORY_OF_PROPERTY` and `NEXT_ENTITY`. It shows that the users performed 101 edits in `Temporal Properties`, and 70 of these edits were followed by the edits in `Diagnostic Method` property on the `same` class. Therefore the confidence of this rule is 69% (i.e., 70 divided by 101).

3.3 Prediction Using Association Rules

Recall that our goal is to predict the next editing operation that a user will make given the current change. The association rules show the relationships between users' next editing operations and the previous edits. To simulate the prediction process, we split our data into two sets: a training set and a test set. We generate the association rules based on the training set and assess the confidence values of these rules in the test set. The difference in the confidence values between these two sets will indicate how much the editing patterns drift. Specifically we evaluate the drift along three dimensions: (1) different stages of ontology development over time; (2) different user groups; and (3) different ontologies.

To split the data based on different group of users, we introduce a method that keeps splitting the data randomly by users into training and testing sets until the two data sets satisfy: 1) They are of roughly the same size. 2) The number of users in the two data sets are roughly the same. The advantage of splitting the data in this way is obvious. First, with two sets with roughly equal

Table 3. Number of data entries in the change log before and after aggregation

No. of change entries	ICTM	ICD-11
Original Data	21,466	180,896
Aggregated on *property category*	6,962	53,908
Aggregated on *property name*	10,208	63,654

size we will have enough data for both training and testing data sets. Secondly, users with different numbers of data entries are randomized into both training and testing datasets so that no bias is introduced due to the splitting process. To split the data based on the time, we divide the data roughly in the middle of the dataset to have equal size of data for both training and testing.

We present the results of this evaluation in the following section.

4 Experimental Results

We have applied the data aggregation process to all ICTM and ICD-11 datasets with five selected features. To show the effect of data aggregation, we list the statistics before and after the data aggregation in Table 3. In both the ICTM and ICD-11 datasets, more than half of the data have been aggregated.

4.1 Rule Analysis for Training Data

All the meaningful rules that we generated from the training data fall into the following three types:

Type One

(a) `CATEGORY_OF_PROPERTY` = `A` $\Longrightarrow$
`NEXT_CATEGORY_OF_PROPERTY` = `B` `NEXT_ENTITY` = `Same`

(b) `NAME_OF_PROPERTY` = `A` $\Longrightarrow$
`NEXT_NAME_OF_PROPERTY` = `B` `NEXT_ENTITY` = `Same`

Type Two

(a) `CATEGORY_OF_PROPERTY` = `A` $\Longrightarrow$
`NEXT_CATEGORY_OF_PROPERTY` = `A` `NEXT_ENTITY` = `Not the same`

(b) `NAME_OF_PROPERTY` = `A` $\Longrightarrow$
`NEXT_NAME_OF_PROPERTY` = `A` `NEXT_ENTITY` = `Not the same`

Type Three

(a) `CATEGORY_OF_PROPERTY` = `A` $\Longrightarrow$
`NEXT_CATEGORY_OF_PROPERTY` = `B` `NEXT_ENTITY` = `Not the same`

(b) NAME_OF_PROPERTY = A $\Longrightarrow$
NEXT_NAME_OF_PROPERTY = B NEXT_ENTITY = Not the same

For each rule type, we show the rules generated when aggregating on the *property category* (rules a), and when aggregating on the *property name* (rules b). *Type One* rules capture the case where the user continues to edit the same class, but changes the property category (1a) that she edits or the property name (1b). The transition means that the following edit occurs in a different tab (1a), or in a different field on the form (1b), respectively (Figure 1). *Type Two* rules describe the situation where the user is focused on editing a single property category (2a) or a single property (2b), e.g., `Short Definition`, for different classes: she edits the property for one class and then edits the same property for another class. *Type Three* rules describe the user who edits both in a different entity and a different property category (3a) or property (3b) in the next operation.

In the rest of this section, we show the top five association rules for ICD-11 and ICTM datasets, aggregated on property category.

Example Association Rules from the ICTM Data. Specifically, Table 4 lists the top five association rules generated from the ICTM data aggregated on category of property. We rank the rules by the confidence measures. For example, rule 1 states that after editing property category `Title & Definition`, users will, with probability of 77% (i.e., 2632 divided by 3409), edit the same property category (`Title & Definition`) but on a different entity. The rest rules are interpreted in a similar way. Rule 2 and rule 3 (*Type One*) show that after editing property category `Temporal Property` or `Causal Property`, users likely continue editing the same class, transitioning to property category `Diagnostic Method`. Rule 1 and rule 4 (*Type Two*) indicate that users will keep editing on the same category of property in the next operation even they transit into another entity. These rules show that the editing tasks of these categories of properties usually come as a batch work. During certain period of time users will edit a set of `Title & Definitions` or `Classification Properties` on different entities.

Example Association Rules from the ICD-11 Data. Table 5 lists the top five generated association rules from the ICD-11 data aggregated on category of property. Again, we rank them by the confidence measures. We found that the editing patterns in ICD-11 are different from the ones in ICTM. The first four rules are *Type Two* rules and the fifth rule is a *Type Three* rule. There are no *Type One* rules in this set. The first four rules show that the users of ICD-11 are very likely to keep editing on the same category of property in the next operation even they transit into another entity. Only rule 5 shows that users may change from editing `Diagnostic Method` to `Title & Definition` while they transit into another entity.

Table 4. Top 5 Association Rules from the ICTM Data

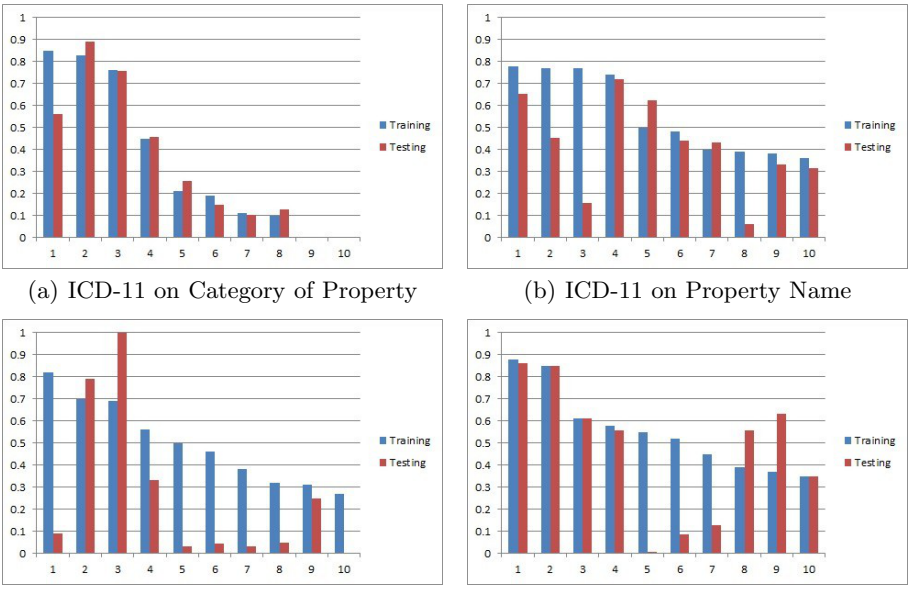
1. CATEGORY_OF_PROPERTY=Title & Definition 3409 ⟹ NEXT_CATEGORY_OF_PROPERTY=Title & Definition NEXT_ENTITY=Not the same 2632 conf:(0.77)
2. CATEGORY_OF_PROPERTY=Temporal Properties 101 ⟹ NEXT_CATEGORY_OF_PROPERTY=Diagnostic Method NEXT_ENTITY=Same 70 conf:(0.69)
3. CATEGORY_OF_PROPERTY=Causal Properties 369 ⟹ NEXT_CATEGORY_OF_PROPERTY=Diagnostic Method NEXT_ENTITY=Same 205 conf:(0.56)
4. CATEGORY_OF_PROPERTY=Classification Properties 1413 ⟹ NEXT_CATEGORY_OF_PROPERTY=Classification Properties NEXT_ENTITY=Not the same 673 conf:(0.48)
5. CATEGORY_OF_PROPERTY=Diagnostic Method 447 ⟹ NEXT_CATEGORY_OF_PROPERTY=Classification Properties NEXT_ENTITY=Not the same 170 conf:(0.38)

Table 5. Top 5 Association Rules from the ICD-11 Data

1. CATEGORY_OF_PROPERTY=Classification Properties 16794 ⟹ NEXT_CATEGORY_OF_PROPERTY=Classification Properties NEXT_ENTITY=Not the same 14772 conf:(0.88)
2. CATEGORY_OF_PROPERTY=Clinical Description 1951 ⟹ NEXT_CATEGORY_OF_PROPERTY=Clinical Description NEXT_ENTITY=Not the same 1639 conf:(0.84)
3. CATEGORY_OF_PROPERTY=Editorial Information 4214 ⟹ NEXT_CATEGORY_OF_PROPERTY=Editorial Information NEXT_ENTITY=Not the same 3430 conf:(0.81)
4. CATEGORY_OF_PROPERTY=Title & Definition 23658 ⟹ NEXT_CATEGORY_OF_PROPERTY=Title & Definition NEXT_ENTITY=Not the same 17993 conf:(0.76)
5. CATEGORY_OF_PROPERTY=Diagnostic Method 447 ⟹ NEXT_CATEGORY_OF_PROPERTY=Title & Definition NEXT_ENTITY=Not the same 157 conf:(0.35)

4.2 Prediction Results on the Testing Data

We apply the association rules generated from to training data to the testing data to simulate the prediction process. If more than 10 meaningful rules are generated from the training data, we report top 10 rules based on the measure of confidence. We calculate the confidence values of these rules in the testing data compared with the original confidence values in the training data. The difference in the confidence values between these two sets will indicate how much the editing patterns drift.



(a) ICD-11 on Category of Property (b) ICD-11 on Property Name

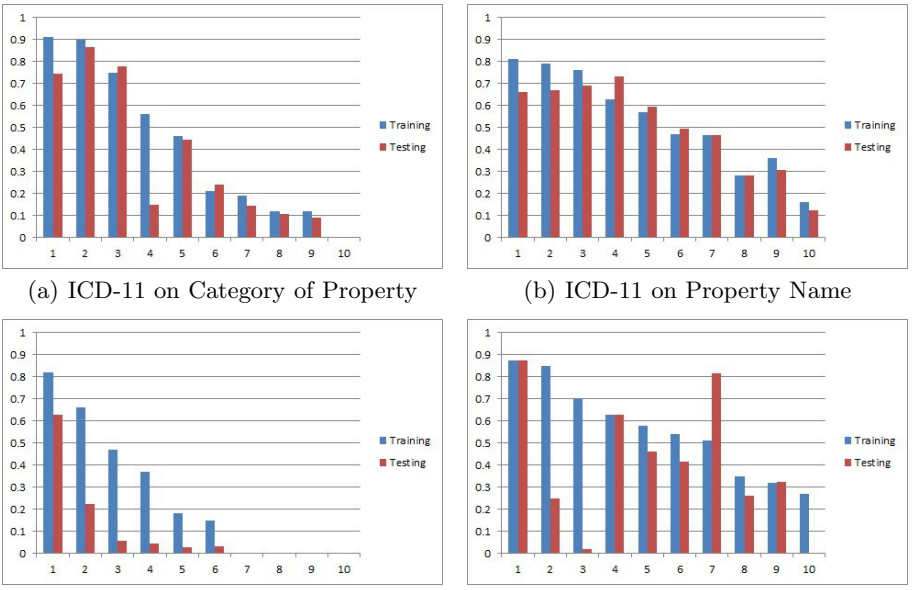
(c) ICTM on Category of Property (d) ICTM on Property Name

Fig. 3. Prediction Across User Groups

Prediction across User Groups. Figure 3 shows a set of prediction results measured by the confidence values from the training and testing data of ICTM and ICD-11. We can see that the results from the ICD-11 data have good prediction accuracy (i.e., the similarity of confidence values from the training and testing data) and are better than the results from the ICTM data. The prediction results from the ICTM data aggregated on the property name are better than results from the data aggregated on the category of property. It shows the users in ICD-11 have similar editing patterns.

Prediction across Time. In Figure 4, we show the prediction results when we split the training and testing data based on time. We can see that the results from the ICD-11 data aggregated on category of property or property name, and the results from the ICTM data aggregated on the property name have good prediction accuracy. The results from the ICTM data aggregated on the category of property only have 6 rules and the prediction results are not good.

Prediction across Ontologies. We also report the prediction results across ontologies (Figure 5). There are two scenarios in our study: 1) We use the ICD-11 data as the training data and the ICTM data as the testing data. 2) We use the ICTM data as the training data and the ICD-11 data as the testing data. We can see in Figure 5 that prediction results from the data aggregated



(a) ICD-11 on Category of Property

(b) ICD-11 on Property Name

(c) ICTM on Category of Property

(d) ICTM on Property Name

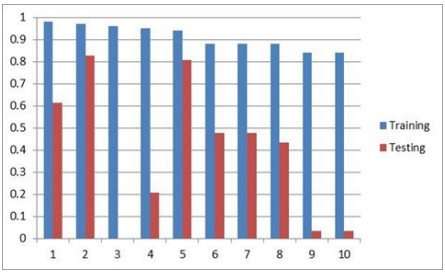
Fig. 4. Prediction Across Time

on property name are still better than the results from the data aggregated on category of property. On the other hand, using ICTM to predict ICD-11 based on the data aggregated on property name is a little better than using ICD-11 to predict ICTM. It may be explained that ICTM use the property names which have been used in ICD-11. Prediction across ontologies might not be as accurate as the ones from across time and across user groups, however they still share plenty of similarities especially on top frequent patterns.

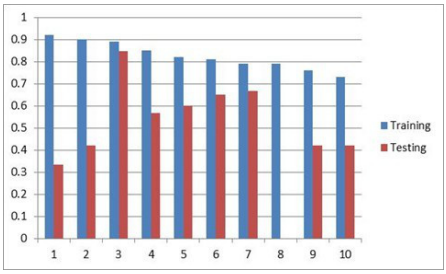
5 Discussion

Our data shows mixed predictive power of association rules. However, the difference in the patterns is in itself useful in analyzing how ontology editing changes from one ontology to another and through different stages of the life cycle. Recall that our key motivation for this work was to find editing patterns so that we can custom-tailor the user interface in order to direct the users' attention to the areas of class definitions that they are likely to edit next.

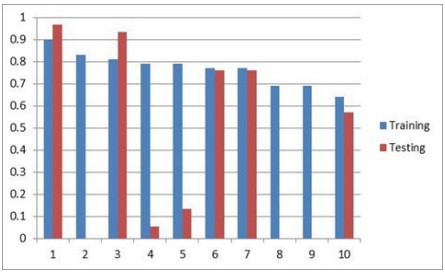
In general, rules that are based on property name rather than property category appear to be more predictive, regardless of whether we look across different users (Figure 3) or different time spans (Figure 4). In other words, the users' transitions between categories of properties are less consistent across the training and test data than their transitions across specific properties. Indeed, a closer look at the data reveals that in the case of ICTM, patterns were particularly



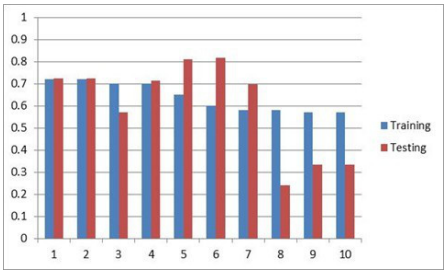
(a) Training on ICD-11 and Testing on ICTM (Category of Property)



(b) Training on ICD-11 and Testing on ICTM (Property Name)



(c) Training on ICTM and Testing on ICD-11 (Category Of Property)



(d) Training on ICTM and Testing on ICD-11 (Property Name)

Fig. 5. Prediction Across Ontologies

different. For example, only two property category (`Title & Definition` and `Classification Properties`) account for almost 90% of the training data in both cases. Thus, half of the users edited only these two property categories, and did so in the beginning of the observation period.

Similarly, while prediction on property names was noticeably better, it is informative to look at the outliers. Consider Figure 3(b), rule 3 in ICD-11: this rule had very high confidence in the training data and was not useful in the testing data. This rule involves a very specialized property, `Primary TAG`, a property describing which group of editors is responsible for the definition of the class. It is likely that the editors would fill out this property only at a particular stage in the lifecycle, and not return to it later. We also selected the association rules based on the measure of support. Comparing the results from rules selected by confidence, the prediction results are similar.

Another reason for better prediction results on property names compared to property categories could be because we had more data in the latter case: consecutive edits on different properties in the same category (a frequent editing pattern) were aggregated in the data preprocessing step (Table 3). In general, the more training data we have, the more reliable the data mining results are.

For the same reason, ICD-11 results show better predictive value than ICTM results: we had considerably more data for ICD-11. It will be interesting to see,

as we get more change logs from the users, whether or not the prediction accuracy for ICTM improves as well.

We have also observed that in cross-ontology prediction, ICTM rules were better predictors for ICD-11 rules than the other way around. Indeed, because our data capture the earlier stages of the ICTM development lifecycle, the ICTM editors focused on the more basic properties, and only occasionally ventured into the more advanced properties. Thus, the rules capturing the basic properties carry over well to ICD-11, but there is not enough data—and the patterns are not yet established—for the other properties.

The rules that we identified have given us important feedback on how we can improve the user interface to support the users' editing patterns better. For instance, we have seen that both in ICTM (Table 4, rules 1 and 4), and especially in ICD-11 (Table 5, rules 1-4), users are editing the same property category over and over again, but in different classes. This rule means that the we can improve the editing experience, if our user interface will preserve the same tab when the user switches to a different class. Furthermore, we have identified that the users are editing the same property for different classes very often. The predominance of this type of rule indicates that we should support a tabular type of user interface that makes it easier for users to edit the same property for different classes. For example, a spreadsheet-like tabular interface could contain in each row a column for the class, another for its title and a third one for its definition. This type of interface would very likely speed the data entry and support the editing patterns we have identified in a data-driven way.

The key lesson from the previous observations is the need for including in the analysis not only the change data but also the data on the lifecycle of the ontology and the roles of the user. In our earlier work, we demonstrated that it is possible to distinguish different user roles by analyzing the change data [5]. Integrating these two analyses will likely produce better predictions. For example, we can analyze the change data for each user individually, or for a set of users with the same role, and use data mining on this subset to predict what that particular user is likely to do. Similarly, accounting for the distribution of the features themselves in the data—and the changes in this distribution—will enable us to capture yet another key aspect of changing logs.

6 Conclusions

In this paper, we analyzed the user editing pattern in ontology development projects with the help of data mining algorithms, specially association rule mining. The experiment result shows that the patterns we generated from the ontology editing history data provide useful and straightforward patterns that could help design a better ontology-editing software by focusing the user attention on the components of the complex class definition that the user is likely to edit next. We can use the discovered editing patterns to develop a recommendation module for our editing tools, and to design user interface components that are better fitted with the user editing behaviors.

In order to achieve better predictive power in the data mining, future analyses must also account for different stages in the ontology life cycle, changing user roles, and, potentially, other components. However, our initial results reported in this paper point the way to the data-driven development of user interfaces that alleviates the cognitive load of complex tasks such as ontology editing for domain experts.

Acknowledgments. This work was supported in part by grants GM086587, EB007684, and GM103309 from the US National Institutes of Health. We thank Margaret Qian for her inputs to this project and this paper.

References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: ACM SIGMOD International Conference on Management of Data, pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: International Conference on Very Large Data Bases, pp. 487–499 (1994)
3. Cosley, D., Frankowski, D., Terveen, L., Riedl, J.: Suggestbot: Using intelligent task routing to help people find work in wikipedia. In: International Conference on Intelligent User Interfaces, pp. 32–41 (2007)
4. De Leenheer, P., Debruyne, C., Peeters, J.: Towards social performance indicators for community-based ontology evolution. In: Workshop on Collaborative Construction, Management and Linking of Structured Knowledge at the International Semantic Web Conference (2009)
5. Falconer, S.M., Tudorache, T., Noy, N.F.: An analysis of collaborative patterns in large-scale ontology development projects. In: International Conference on Knowledge Capture, pp. 25–32 (2011)
6. Gibson, A., Wolstencroft, K., Stevens, R.: Promotion of ontological comprehension: Exposing terms and metadata with web 2.0. In: Workshop on Social and Collaborative Construction of Structured Knowledge (2007)
7. GO Consortium: Creating the Gene Ontology resource: design and implementation. Genome Res. 11(8), 1425–1433 (2001)
8. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations 11(1), 10–18 (2009)
9. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers (2001)
10. Hartung, M., Kirsten, T., Gross, A., Rahm, E.: Onex: Exploring changes in life science ontologies. BMC Bioinformatics 10, 250 (2009)
11. Malone, J., Stevens, R.: Measuring the level of activity in community built bio-ontologies. Journal of Biomedical Informatics 46(1), 5–14 (2013)
12. Noy, N.F., Chugh, A., Liu, W., Musen, M.A.: A framework for ontology evolution in collaborative environments. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 544–558. Springer, Heidelberg (2006)
13. Pesquita, C., Couto, F.M.: Predicting the extension of biomedical ontologies. PLoS Computational Biology 8(9) (2012)

14. Pöschko, J., Strohmaier, M., Tudorache, T., Noy, N.F., Musen, M.A.: Pragmatic analysis of crowd-based knowledge production systems with icat analytics: Visualizing changes to the icd-11 ontology. In: AAAI Spring Symposium on Wisdom of the Crowds, pp. 59–64 (2012)
15. Rector, A.L., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H., Wroe, C.: OWL pizzas: Practical experience of teaching OWL-DL: Common errors & common patterns. In: International Conference on Knowledge Engineering and Knowledge Management, pp. 63–81 (2004)
16. Sebastian, A., Noy, N.F., Tudorache, T., Musen, M.A.: A generic ontology for collaborative ontology-development workflows. In: International Conference on Knowledge Engineering and Knowledge Management, pp. 318–328 (2008)
17. Sioutos, N., de Coronado, S., Haber, M., Hartel, F., Shaiu, W., Wright, L.: NCI Thesaurus: A semantic model integrating cancer-related clinical and molecular information. Journal of Biomedical Informatics 40(1), 30–43 (2007)
18. Strohmaier, M., Walk, S., Pöschko, J., Lamprecht, D., Tudorache, T., Nyulas, C., Musen, M.A., Noy, N.F.: How ontologies are made: Studying the hidden social dynamics behind collaborative ontology engineering projects. Journal of Web Semantics 20, 18–34 (2013)
19. Tudorache, T., Falconer, S., Nyulas, C., Noy, N.F., Musen, M.A.: Will semantic web technologies work for the development of ICD-11? In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 257–272. Springer, Heidelberg (2010)
20. Tudorache, T., Nyulas, C., Noy, N.F., Musen, M.A.: WebProtégé: A collaborative ontology editor and knowledge acquisition tool for the web. Semantic Web Journal 4(1), 89–99 (2013)
21. Walk, S., Pöschko, J., Strohmaier, M., Andrews, K., Tudorache, T., Noy, N.F., Nyulas, C., Musen, M.A.: Pragmatix: An interactive tool for visualizing the creation process behind collaboratively engineered ontologies. International Journal on Semantic Web and Information Systems to appear (Special issue on Visualisation of and Interaction with Semantic Web Data) (2013)