Noname manuscript No.
(will be inserted by the editor)

A Method for Detecting Behavior-based User Profiles in Collaborative Ontology Engineering

Sven Van Laere · Ronald Buyl · Marc Nyssen · Christophe Debruyne

Received: date / Accepted: date

Abstract Collaborative ontology-engineering methods usually prescribe a set of processes, activities, types of stakeholders and the roles each stakeholder plays in these activities. We, however, believe that the stakeholder community of each ontology-engineering project is different and one can therefore observe different types of user behavior. It may thus very well be that the prescribed set of stakeholder types and roles do not suffice. If one were able to identify these user behavior types, which we will call a user profile, one can compliment or revisit those predefined roles. For instance, those user profiles can be used to provide customized interfaces for optimizing activities in certain ontology-engineering projects. We present a method that discovers different user profiles based on the interactions users have with each other in a collaborative ontology-engineering environment. Our approach clusters the users based on the types of interactions they perform, which are retrieved from datasets that were annotated with an interaction ontology – built on top of SIOC – that we have developed. We demonstrate our method using the database of two instances of the GOSPL ontology-engineering tool. The databases contain the interactions of two distinct ontology-engineering projects involving respectively 42 and 36 users. For each dataset, we discuss the findings by analyzing the different clusters. We found that we are able to discover different user profiles, indicating that the approach we have taken is viable, though more experiments are needed to validate the results.

Keywords Collaborative Ontology Engineering · User Profiling · Collaboration · Ontology-engineering Tool · Clustering

1 Introduction

An ontology is commonly defined as: "*a [formal,] explicit specification of a [shared] conceptualization*" [13], and is key in enabling semantic interoperability between autonomously developed information systems belonging to a community of stakeholders. Ontology engineering is far from trivial and require adequate methods and tools to support these communities in the ontology-engineering process.

Different methods and tools for collaborative ontology engineering prescribe different roles in the ontology construction process. However, we believe that the different "types" of user behavior – which we will call a *user profile* from now on – are a priori not known as the stakeholder communities differ for each ontology-engineering project. We therefore also believe that the predefined roles and responsibilities should at least be complemented with the user profiles based on their behavior in the complex socio-technical environment they are collaborating in. When a project starts, roles are assigned based on the confidence and reliability a project leader has in a person or the kind of input a stakeholder can provide in the ontology-construction process. Behavior-based approaches use the user's behavior as a model, commonly relying on machine-learning techniques to discover useful patterns in it [25].

People are often assigned to a task groups in an ontology-engineering project where the project leader attributes tasks among different team members. However, for the project leader it can be difficult to label users adequately and to know how users prefer to interact with each other and a tool. An important assumption in this paper is thus: *the user profiles in a community or communities are not known beforehand.* The potential of identifying many different profiles are manifold, including: (i) personalizing the interaction a user has with a system to render their tasks more efficient (adaptive hypermedia), (ii) detect a group of users' expertise and thus provide means to fully exploit that group's capabilities, (iii) composing tasks groups with

S. Van Laere · R. Buyl · M. Nyssen
Vrije Universiteit Brussel, Department of Public Health, Biostatistics and Medical Informatics (BISI) Research Group, Laarbeeklaan 103, 1090 Jette, Belgium
Tel.: +32-2-477-4444, Fax: +32-2-477-4000,
E-mail: {svvlaere,rbuyl,mnyssen}@vub.ac.be

C. Debruyne
ADAPT Centre, Trinity College Dublin, Dublin 2, Ireland
WISE Lab, Vrije Universiteit Brussel, Brussels, Belgium
E-mail: debruync@scss.tcd.ie

a variety of skills and competencies to automate some of the ontology engineering processes.

This paper presents some preliminary results in answering the following research question: "How can we identify user profiles by analyzing the interactions between users, and between users in a collaborative ontology engineering environment?" Besides this research question, we are interested in how we can extract these interactions in a generic way. This paper is organized as follows: Section 2 provides a background in the applications of user profiling and their application to the field of collaborative ontology engineering; Section 3 elaborates on the GOSPL ontology engineering method since we use this to evaluate our method presented later and we focus on the creation of a generic ontology for extracting interactions made. Section 4 gives an overview of the method for user profiling two ontology engineering datasets; Section 5 applies the user profiling algorithm on two ontology engineering datasets and gives an interpretation of the found profiles; In Section 6 we discuss the outcome of the clustering algorithm and conclude the paper.

2 Related Work

User profiling is an emerging research field that considers the application of finding structures in the way people behave. One can derive a lot of information from the social interactions between users, and between a user and a system when observing online communities. People can be classified according to their behavior, taste, effort, etc. Related work on user profiling can be found in different fields: e-commerce [23, 32], computer-supported cooperative work (CSCW) [24, 28], web browsing [21, 22, 33], news feeds [2, 20, 30], etc.

Also in the field of information governance, user profiling is emerging. Gartner defines information governance as: *"the specification of decision rights and an accountability framework to encourage desirable behavior in the valuation, creation, storage, use, archival and deletion of information. It includes the processes [actions], roles [actors], standards and metrics [actands] that ensure the effective and efficient use of information in enabling an organization to achieve its goals."*[1] Most scientific papers are directly inspired by traditional data quality management and IT governance [18], and propose deterministic role patterns and decision domains with a predefined terminology. Yet, it is necessary that these models need to be flexible at runtime, i.e. contingent upon issues [6]. Here, our approach can help by making the process more generic.

When we investigate these topics it is key to have a clear understanding of what roles are, how they relate to human behaviors, and how these behaviors can be captured in terms of online community features. A discussion about the definition of a role can be found in [12]. In their discussion they state that a user role can arise either from the social context of a person and the dynamics of their relationships or from repeated interactions and agreements across practices. In this work, we adopt the second definition of role. Usually a set of behavioral dimensions is used to distinguish user profiles; here we use types of interactions. Examples of roles mentioned in the literature are: newbies, experts or lurkers [31].

Each of these roles is identified by a set of behaviors, (or behavioral dimensions), such as engagement, contribution, popularity, participation, etc. The general procedure to model behavior in an online community is by translating them into measurable behavioral features from the social network graph with an associated intensity level (see e.g. [16, 26]). *In contrast we will use features based on the different interactions between users and between the user and the system.*

This study focuses on applying data mining techniques for extracting user profiles in ontology engineering. Though not much related can be found in the field of ontology engineering, De Leenheer *et al.* aimed at relating performance indicators with "user types", and therefore applied simple statistical measures for classifying users [6]. Falconer *et al.* described how they sought and discovered collaboration patterns in the creation of ontologies in the medical domain described how they sought and discovered collaboration patterns in the creation of ontologies in the medical domain [11]. Indirectly, those collaboration patterns provide insights about the "user types" in that experiment. The identification of these collaboration patterns provides input on thow to facilitate the activities performed by these users.

Other related work is concerned with predicting the next editing operation a user is likely to make in a collaborative ontology-engineering environment using association use mining [37, 36] and Markov chains [35]. Those studies were conducted to investigate how these ontologies evolve and what the editing sequences were. Those insights can be used to render certain tasks or activities more efficient by providing the users customized interfaces. Finally, the work presented in [9] explores the potential identifying "community leaders" via a reputation system that monitors certain interactions (between users and between a user and the system). Their

[1] `http://blogs.gartner.com/debra_logan/2010/01/11/what-is-information-governance-and-why-is-it-so-hard/` (last accessed on January 24, 2016)

work is declarative and focuses on the identification of those so-called community leaders.

3 Social Interactions in Ontology Engineering

Ontologies are a social artifact as they are built for a purpose and the result of agreement processes within the stakeholder community. These agreement processes can be broken down into different interactions take place in collaborative ontology engineering as proposing new concepts and relations, discussing proposals, voting, taking decisions and so forth. Though there quite a few collaborative ontology engineering methods: HCOME [19], DILIGENT [27], BSM [7], etc., not all of them have proposed or provided a tool (e.g. [8]). Similarly, there are tools available which support collaborative ontology engineering, but are not explicitly tailored to support a specific ontology-engineering method. One example of such a tool is WebProtégé [34].

Not all tools capture the social processes that drive the ontology-construction process [10] and in order to test our hypothesis that one can observe different types of user behavior in collaborative ontology-engineering projects, we should have access to datasets that captured these interactions.

For this study, we had access to the databases of two ontology-engineering projects using the GOSPL method and tool [10]. GOSPL will be described in more detail in the next section, but we would first like to stress that our method is generic enough to be applicable in other tools where these social processes are logged. The generality of our approach will be ensured by the creation of an ontology for social processes – which will be used to distill the dataset from the databases – that can be extended for each method.

Before elaborating on the ontology for social processes and the extraction of the dataset, we will first elaborate on the GOSPL method. Though our method for discovering user profiles is generic, the remaining sections of this paper will describe the results of our method using datasets from ontology-engineering projects with GOSPL. A notion of the GOSPL ontology-engineering method is thus necessary to understand the remainder of this paper.

3.1 GOSPL: Grounding Ontologies with Social Processes and Natural Language

For this paper, we had access to two databases of two ontology-engineering projects using GOSPL. GOSPL is a community-based collaborative ontology-engineering method which formalized the social processes and where the description of the Universe of Discourse does not depend on the language the ontology will be implemented in (such as OWL).

In GOSPL, concepts are both represented formally and informally. Formally by means of *lexons* and informally by means of *glosses*, which are definitions in natural language. An example of a lexon is <Cultural Domain, Concert, is a, subsumes, Event>, which states that, in the “Cultural Domain” community, the concept referred to with term “Concerts” plays the role of “is a” on the concept with term “Event”, and the concept with term “Event” plays the role of “subsumes” on the concept with term “Concert”. A lexon should ideally result in two meaningful[2] sentences when read in both directions, but assuring this quality is the responsibility of the community.

Synonyms are agreements that two community-term pairs refer to the same concept, and gloss-equivalences are agreements that two glosses refer to the same concept. Ideally, if two communities agree that two glosses refer to the same concept, the labels associated with those glosses should be considered equal as well; this is called the glossary consistency principle [10]. This allows communities to agree on the “sameness” of both types of representation at different times in the ontology-engineering process and have this principle drive the discussions within the communities to revisit their ontologies to ensure consistency.

Fig. 1 depicts the different processes in GOSPL. Starting from co-evolving communities and requirements, the informal descriptions of key terms have to be gathered before formally describing those concepts. Communities define the semantic interoperability requirements, out of which a set of key terms is identified. Those terms need to be informally described before the formal description can be added. In order for a lexon to be entered, at least one of the terms needs to be articulated first. The terms and roles in lexons can be constrained and the community can then commit to the hybrid ontology by annotating an individual application symbols with a constrained subset of the lexons. At the same time, communities can interact to agree on the equivalence of glosses and the synonymy of terms. Committing to the ontology allows for the data to be explored by other agents via that ontology. Commitments also enable the community to re-interpret the ontology with its extension (i.e. the instances in each annotated system). This will trigger new social processes that lead to a better approximation of the domain, as the community is able to explore the increasingly annotated data, e.g., by formulating queries.

[2] That is, meaningful for that community.

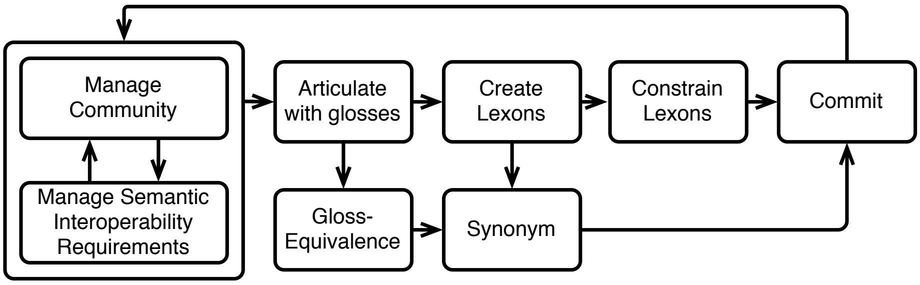

Fig. 1 The different phases in the GOSPL ontology-engineering method. Image from [10].

3.2 Creation of an Ontology and Annotation of the Data

We developed a first OWL ontology[3] – built on top of FOAF[4], SIOC[5] and Dublin Core[6] – to capture the notion of "interaction" in collaborative environments. That ontology was then imported in a second OWL ontology to declare the different social processes in GOSPL. Both ontologies are then used to annotate the databases of the GOSPL tool to create our datasets, which will take on the form of an RDF [14] knowledge base with RDB-to-RDF technologies such as D2RQ [3] or R2RML [5].

Fig. 2 depicts the first ontology, which introduced the concept "Interaction" and declares has sioc:Post as a subclass of that concept. This is because we deem every post on any kind of social and collaborative platform as an interaction. Not all interactions are posts, however. One example of an interaction that is not a post is voting, which is furthermore a concept that was part of the GOSPL tool as will be described later on.

The adoption of Semantic Web technologies such as OWL is motivated by the fact that it allows our framework to retrieve and reason about the available social processes of different social platforms for user profiling. In other words, the use of ontologies – with notions such as Interaction and Post (from SIOC) – renders our framework more generic.

Knowledge bases using this ontology can now retrieve all instances of interactions as well as retrieve all types of interactions in two ways: via query simple query (provided reasoning is enabled or all inferences have been materialized in the knowledge base prior to querying) or via a SPARQL 1.1 query using property paths. Examples of both are shown in Table 1.

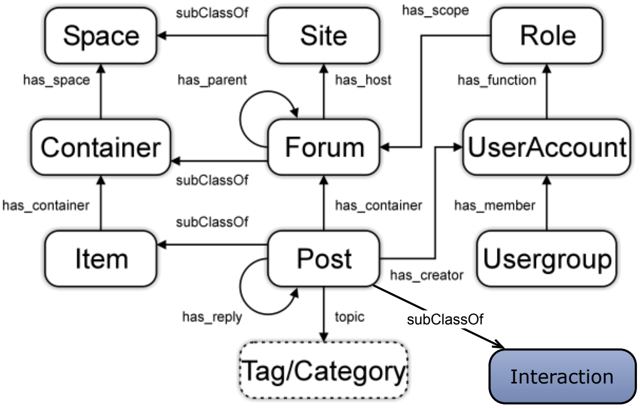

Fig. 2 The interaction ontology introducing the concept of "Interaction", which is subclassed by `sioc:Post`.

Table 1 Two queries for retrieving the different types of interaction in a knowledge base. The first assumes that reasoning is enabled; in the second query, inferences have already been materialized in the knowledge base. The latter requires SPARQL 1.1 support. Prefixes are omitted.

Query I

```
SELECT DISTINCT ?interaction WHERE {
  ?interaction rdfs:subClassOf ont:Interaction.
}
```

Query II

```
SELECT DISTINCT ?interaction WHERE {
  ?interaction rdfs:subClassOf+ ont:Interaction.
}
```

Ontology Design Patterns (ODPs) are, according to [4]: *"a reusable successful solution to a recurrent modeling problem."* Several types of ODPs exist, but Content ODPs are small ontologies that cover a limited set of requirements – formulated as competency questions – for a specific problem. The first ontology is actually a Content ODP allowing one to answer the following competency questions:

1. What are the interactions in a collaborative environment?

[3] The ontology can be found on `http://minf.vub.ac.be/ODBASE/interactions.rdf`

[4] `http://www.foaf-project.org/`

[5] `http://sioc-project.org/`

[6] `http://dublincore.org/`

2. What are the types of interactions in a collaborative environment?

The ontology we developed[7], captures the social interactions in a hierarchical manner. A first distinction we could make in this ontology is the difference between social processes to start a discussion within one community (e.g. a request to add a lexon) and social processes to start a discussion between communities (e.g. requests to add synonyms). We call such processes respectively intra- and inter-community requests. Both are called requests when the distinction does not need to be made. Members of a community can propose changes to the ontology (both formal and informal part) in a forum-like manner and encourages the other stakeholders to express their opinion. The stakeholders can do this by replying (to the proposition or to another reply). Another way for community members to express their opinion is to cast a "vote" in the tool. Thus, next to two types of requests, we also have, replies and votes. Eventually discussions can also be closed by concluding the discussion.

This call is not made by a kind of "super user", but by the community in which the discussion takes place. A member of the community asks to all other users to vote whether or not they agree with the initial proposed request within a certain time frame. That is a reason for also taking the closing of a topic into consideration as a social interaction.

During the development of the ontology, we have decided to create subclasses of intra-community requests to relate all requests for a particular part of the ontology (gloss, lexon, constraint, etc.). This hierarchy will later come in handy for feature selection for the mining algorithm.

3.3 Grouping the Social Interactions

In this study, we adopted unsupervised learning by applying the K-means mining algorithm. The choice for an unsupervised learning algorithm is motivated by our assumption that user types are a priori not known. A first step in mining is the choice of attributes and data pre-processing. Since the number of features that are chosen should ideally not be too big with respect to the two data sets mentioned in Section 5, we choose to group the social interactions according to the phases in GOSPL. For instance, we grouped all social interactions related to lexons. The result of this grouping leads to a hierarchy of social interactions. A part of this "grouping" of requests is depicted in Fig. 3.

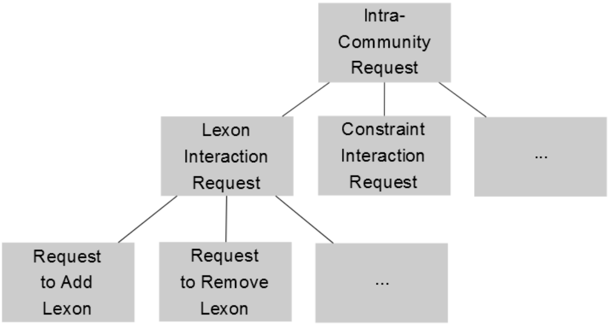

Fig. 3 Graphical depiction of a part of the "grouping" of requests.

4 User Profiling on the Dataset

We apply our ideas on the database of the GOSPL tool for role identification and role composition analysis. The GOSPL tool provides a collection of online forums (called communities) in which stakeholders are working together to engineer an ontology. Users post requests to the system on which other users can reply in a natural language what they think about the proposed request. To express their opinion, people can vote indicating whether they (dis)liked the proposed request. Based on the outcome of the replies, votes and discussions, one of the stakeholders will close the request by accepting or rejecting it. The result of these interactions will eventually lead to a shared ontology.

We were provided with two datasets of the GOSPL tool. The data stems from two small ontology engineering projects involving respectively 42 and 36 users. These people were asked to work together to develop an ontology about a common concept using the GOSPL tool. Unlike programming fora (see e.g. [31]), where people not necessarily working together can ask questions, we are working here with a more closed setting since we are looking in the field of ontology engineering. Here it is key to find agreements. In the case stakeholders want to propose a new request, they will just start a new thread; that is why e.g. threads are not replied on very often. We can distinguish three different phases in this user profiling process (see Fig. 4). In the extraction phase we extract the interactions which will be used as the basis for our profiles. In the manipulation phase we apply statistical techniques in order to prepare the data for the final phase where we apply clustering. This last phase is called the clustering phase.

[7] The ontology can be found on `http://minf.vub.ac.be/ODBASE/gosplinteractions.rdf`

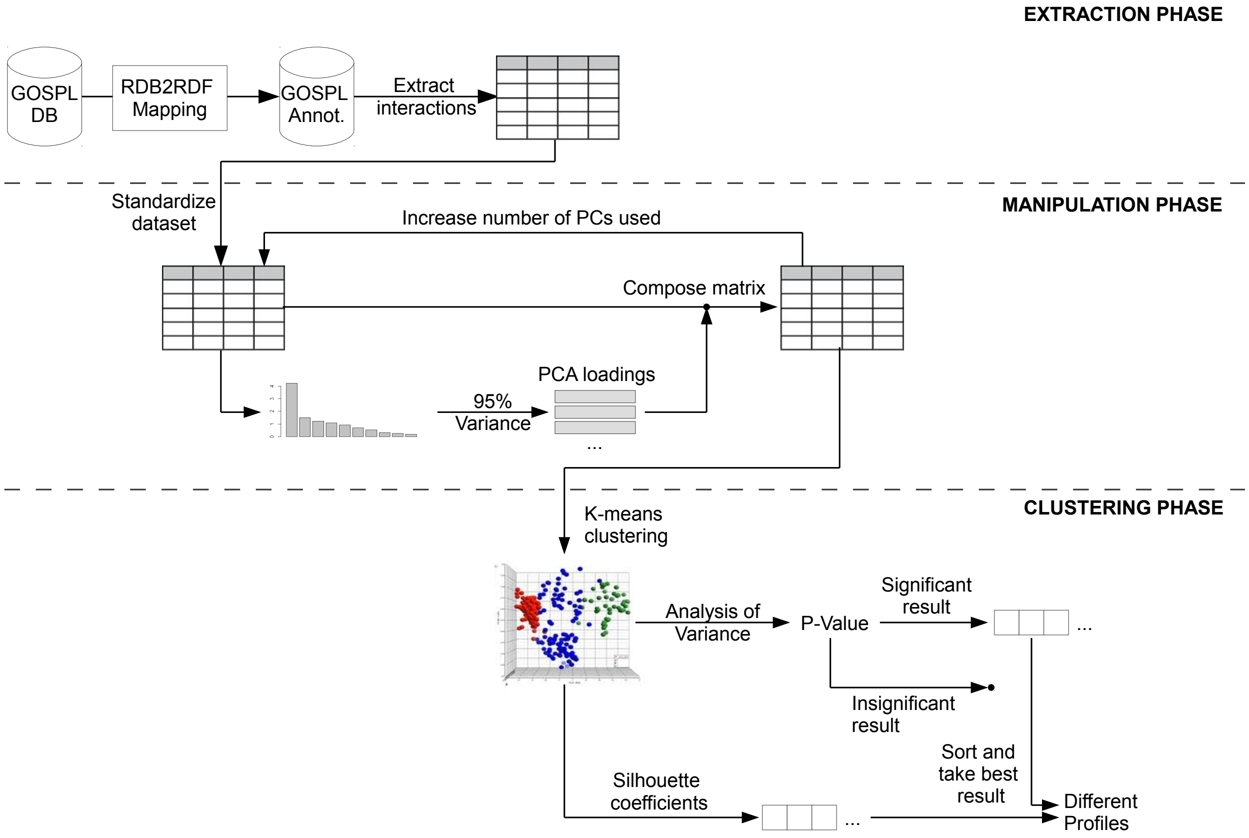

Fig. 4 Graphical depiction of the process to distinguish different types of users

4.1 Extraction phase

Contrary to what most user-profiling applications use, we are only interested in the type of interactions a user is involved in. That is why in a first phase it is necessary to extract these interactions from the two GOSPL databases. Here we will create an RDF knowledge base by populating the ontology for each database with an appropriate mapping with RDB-to-RDF technologies. This allows us to query these knowledge bases with SPARQL [29].

Important here is that since GOSPL has well over 20 types of social interactions [10], the number of features should ideally not be too big with respect to the number of users. In our case the number of users is respectively 42 and 36. That is why we grouped the social processes according to the phases of the method GOSPL. For instance, we grouped all social interactions related to lexons (see e.g. Figure 3).

After we extract the interactions per user, we first standardize the data. The reason for standardization is that in a later phase, we will execute a principal component analysis (PCA) [17] that projects data onto directions which maximize the variance. In order to be able to compare variances across different features we standardize the original extracted data.

4.2 Manipulation phase

In this phase we manipulate the data in order to transform it using a principal component analysis (PCA). In the PCA analysis we look at how many principal components (PCs) are needed to cover a total cumulative variance of 95% (see e.g. [17]).

Once we have found this number N, we compose a new matrix by multiplying the standardized data by the found PCA loadings (see Algorithm 1). We iteratively raise the number of PCs used to transform the dataset ranging from 2 to N. In each iteration we will cluster based upon the newly transformed matrix consisting of the same amount of users, but a reduced amount of dimensions.

Algorithm 1 Transform data by applying PCA after standardization

```
procedure TRANSFORM($data$)
    $standardizedData \leftarrow Standardize(data)$
    $pca \leftarrow ExecutePCA(standardizedData)$
    $loadings \leftarrow GetRotations(pca)$

    $transformedData \leftarrow (standardizedData \times loadings)$

    return transformedData
end procedure

procedure STANDARDIZE($data$)
    $dimensions \leftarrow Columns(data)$
    $users \leftarrow Rows(data)$

    for $i \leftarrow 1, users$ do
        for $j \leftarrow 1, dimensions$ do
            $newData[i, j] \leftarrow \frac{Data[i,j]-\mu_j}{\sigma_j}$

    return $newData$
end procedure
```

4.3 Clustering phase

In this last phase we use the transformed dataset, after applying standardization and PCA, to cluster the different users based on their interaction behavior. We apply the K-means clustering algorithm on this dataset. Since we do not know the unknown variable K (number of clusters to be found), which is needed for clustering, we iteratively raise K (see Algorithm 2) from 2 to $\lfloor\sqrt{n}\rfloor$, where n represents the number of users. These cluster results will then be evaluated based on the outcome of an analysis of variance (ANOVA) [1] with a significance level of $\alpha = 0.05$ and an evaluation of the silhouette coefficient belonging to that cluster result.

By performing an ANOVA we test if there is a statistical significant different ($p < \alpha$) between different user groups (profiles) for each of the principal components used to represent the dataset. We will explain this by example: Suppose we work with a reduced dataset of two PCs and $K = 2$. Then we will first perform the ANOVA for the first PC using the groups of the clustering and look if we have a statistical result that is significant. Secondly, we will also do this for the second PC. If both ANOVA tests are significant, this result in the end will be taken into account after calculating the silhouette coefficients. If one or both tests are insignificant, we will not take this result into account.

We calculate silhouette coefficients to have a measure for internal homogeneity and external heterogeneity between the different clusters. This silhouette coefficient produced is calculated as follows:

$$s_i = \frac{b_i - a_i}{max(a_i, b_i)}, \tag{1}$$

Where a_i denotes the average distance to all other items in the same cluster, and b_i is given by calculating the average distance with all other items in each other distinct cluster and then taking the minimum distance. The value of this silhouette coefficient s_i ranges between -1 and 1, where the former indicates a poor clustering where distinct items are in the same cluster and the latter indicates perfect cluster cohesion and separation. The coefficient thus provides a quality measurement for the cluster method based on how similar intra-cluster items are (cohesion) and how dissimilar inter-cluster items are (separation).

In the end we consider the significant ANOVA tests and look which result has the highest silhouette coefficient. This cluster result is then a basis for distinguishing different profiles that can be used in the continuation of an OE project.

Algorithm 2 Look for most significant user profiling

```
procedure BESTUSERPROFILING($data$)
    $highestSilhouette \leftarrow 0$
    $sigUserPr \leftarrow Array(k, princomp)$

    $users \leftarrow Rows(data)$
    $maxPCs \leftarrow InterestingPCs(data)$
    $maxK \leftarrow \lceil Sqrt(users) \rceil$
    for $i \leftarrow 1, maxPCs$ do
        $reduced \leftarrow TakeFirstColumns(data, i)$
        for $j \leftarrow 2, maxK$ do
            $clusters \leftarrow PerformKMeans(reduced, j)$
            $aov \leftarrow PerformANOVA(clusters)$
            $significance \leftarrow PValue(aov)$
            if $significance \leq 0.05$ then
                $silhouette \leftarrow CalculateSilhouette(clusters)$
                if $silhouette < highestSilhouette$ then
                    $highestSilhouette \leftarrow silhouette$
                    $princomp \leftarrow i$
                    $k \leftarrow j$

    return $sigUserPr$
end procedure
```

5 User Profiling on the Dataset

As a result of the clustering phase, the algorithm from the previous section provides us with the necessary information to determine how to cluster the user behavior. We present the data here as a matrix where two unknown variables are used, i.e. the number of clusters (variable K) and the number of principal components to be used, respectively represented by rows and columns.

5.1 Dataset I

The first dataset consists of 42 users that produce 9,195 user interactions in total spread over a period of a month. These 42 users worked together to engineer an ontology around the concept *Event* in groups of 4 to 6 users. In a first phase we extract 10 dimensions for these 42 users.

After applying a PCA, we observe that by using 5 principal components of the original 10 we cover a total variance of 95.75%. This means that we can iterate to compose a matrix in the manipulation phase ranging from 2 to 5 dimensions (i.e. principal components). On this composed matrix we apply K-means and observe if the clusters that are found are significant for each of the dimensions. Combined with the silhouette coefficients s_i calculated on the clustering outcome, we result in Table 2. Finally we sort these results based on the calculated silhouette coefficients only considering the significant results.

In this experiment we observe the highest silhouette making use of 2 principal components and $K = 5$. This means we distinguish 5 different user profiles in this dataset.

Table 2 Output of clustering phase for experiment 1. The columns *Sig?* represent if the result is significant (Y) or not (N). The columns s_i represent the silhouette coefficient belonging to the cluster result.

	2 PCs		3 PCs	
	Sig?	s_i	Sig?	s_i
$K = 2$	N	0.8465052	N	0.8211668
$K = 3$	Y	0.4808486	N	0.394155
$K = 4$	Y	0.5063455	Y	0.4376691
$K = 5$	Y	0.5151144	Y	0.4467786
$K = 6$	Y	0.4869484	Y	0.4174907

	4 PCs		5PCs	
	Sig?	s_i	Sig?	s_i
$K = 2$	N	0.7990878	N	0.7919471
$K = 3$	N	0.3447474	N	0.3289679
$K = 4$	Y	0.4064185	N	0.3833227
$K = 5$	N	0.4584918	N	0.4640916
$K = 6$	Y	0.4909751	N	0.3957277

5.2 Dataset II

In a second dataset 36 users of the GOSPL tool cooperated to constitute a common ontology considering the concept *Scientific publications*. In total these users worked together for two month, resulting in 7,127 interactions in total. Like the first dataset, these users worked together in groups of 4 to 6 users. In a first phase we extract 11 dimensions[8] for 36 users. In the manipulation phase, we observed that by using the first 8 principal components we cover 95.37% of the total variance in the dataset. Now we can use these to compose a new matrix in the manipulation phase ranging from 2 to 8 dimensions (i.e. principal components). K-means clustering and silhouette coefficient calculations result in Table 3. After sorting out the *best* result, based on the significance and the silhouette coefficients, we observed that using 2 PCs and parameter $K = 3$ results in the highest silhouette coefficient.

Table 3 Output of clustering phase for experiment 2. The columns *Sig?* represent if the result is significant (Y) or not (N). The columns s_i represent the silhouette coefficient belonging to the cluster result.

	2 PCs		3 PCs	
	Sig?	s_i	Sig?	s_i
$K = 2$	N	0.6479863	N	0.5808876
$K = 3$	Y	0.5469009	N	0.4733644
$K = 4$	Y	0.546263	N	0.41647
$K = 5$	Y	0.5389498	Y	0.495092
$K = 6$	Y	0.5284029	Y	0.4758109

	4 PCs		5 PCs	
	Sig?	s_i	Sig?	s_i
$K = 2$	N	0.5964199	N	0.5724287
$K = 3$	N	0.5004817	N	0.5122605
$K = 4$	N	0.4487667	N	0.4988729
$K = 5$	Y	0.4508551	N	0.4310294
$K = 6$	Y	0.4312709	Y	0.4264906

	6 PCs		7 PCs	
	Sig?	s_i	Sig?	s_i
$K = 2$	N	0.5518503	N	0.5387554
$K = 3$	N	0.4508124	N	0.3132408
$K = 4$	N	0.4734011	N	0.3700723
$K = 5$	N	0.4449335	N	0.3821288
$K = 6$	N	0.3909784	N	0.4202661

	8 PCs	
	Sig?	s_i
$K = 2$	N	0.5273153
$K = 3$	N	0.399256
$K = 4$	N	0.4452684
$K = 5$	N	0.3940165
$K = 6$	N	0.4101321

[8] Note that for the same structured dataset we can now extract 11 dimensions since here the semantic interoperability required is now captured in the GOSPL tool, whereas in the first dataset these requirements were not explicitly captured.

5.3 Interpretation of User Profiles

We presented a method for finding different user profiles, by clustering based on the user interactions. However, what is the meaning of these different types of users? Since we cannot identify the characteristics of these clusters automatically, we should interpret them. This can be done observing differences between the averages of the original data for the different user profiles.

5.3.1 User Profiles in Dataset I

In this dataset we observe 5 distinct profiles in Fig. 5. We interpret them by looking at descriptive statistics of each cluster (see Table 4) and observe the following differences between these user types:

- We interpret *cluster 1* (○ symbol) and *cluster 4* (× symbol) together since both clusters have similar behavior for nearly all of the dimensions. On overall, we consider these two groups as active users of the GOSPL tool. However, we do observe that cluster 4 is more familiar with expressing his personal opinion than cluster 1 by replying on a certain topic or by casting votes, respectively dimensions 8 and 10.
- *Cluster 2* (△ symbol) is less active than the previous mentioned user profiles. However, we do observe this user profile prefers interacting with GOSPL by replying, concluding discussions and casting votes, respectively dimensions 8, 9 and 10.
- *Cluster 3* (+ symbol) exists of only one user in this dataset. This user instance has the maximum amount of user interactions for nearly each dimension. In the plots we observe this instance behaves extremely different than the others.
- *Cluster 5* (◇ symbol) can be considered as the more passive group of users, since this user profile has the lowest amount of user interactions.

5.3.2 User Profiles in Dataset II

In dataset II we observed 3 different types of users in Fig. 6. We interpret them by looking at descriptive statistics of each cluster (see Table 5) and observe the following differences between these two user types:

- *Cluster 1* (△ symbol) has a higher amount of interactions as the other two user profiles for more than half of the dimensions. We consider this cluster as being the active users.
- *Cluster 2* (○ symbol) and *cluster 3* (+ symbol) both have a lower interaction amount than cluster 1. However, we consider them as separate groups of users since cluster 2 has in general[9] more interactions than cluster 3. Note that cluster 2 is very active in making interactions considering lexons.

[9] Dimensions 4, 5, 6 and 7 have very low interaction amounts over the complete dataset.

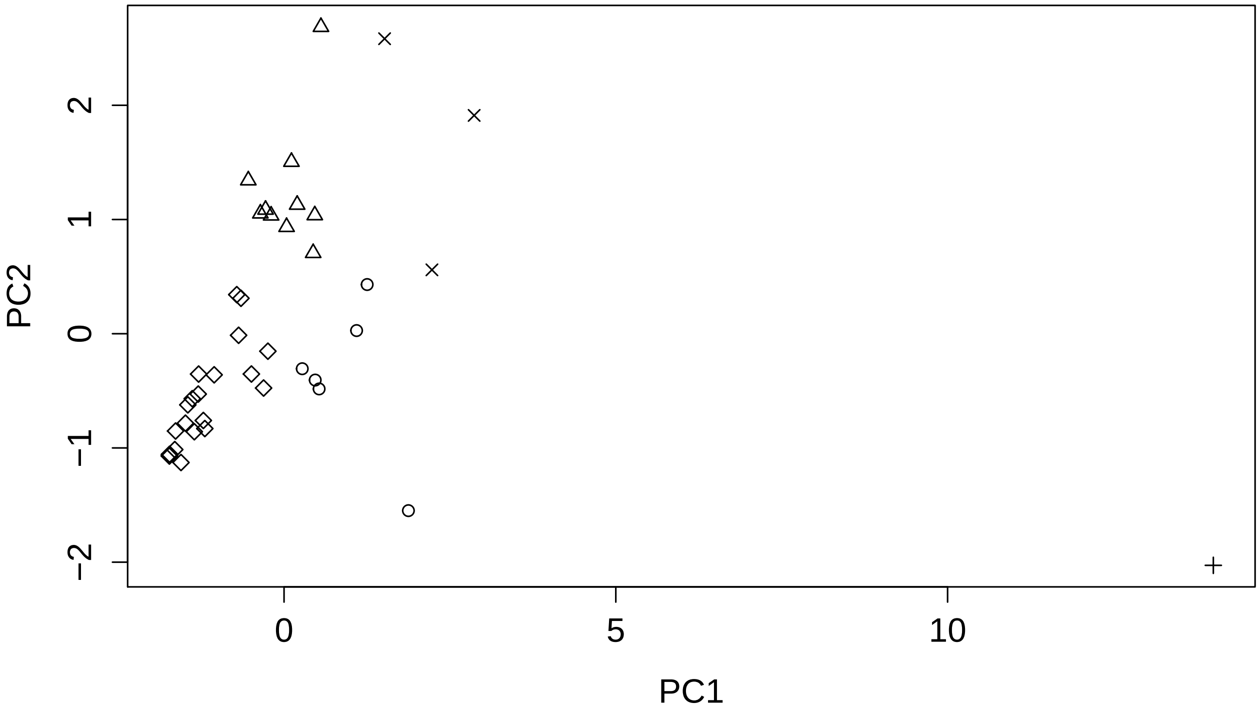

Fig. 5 Clusters of dataset I (clusters indicated with symbols)

6 Discussion and Conclusion

In literature the identification process of different profiles often makes use of both quantitative and qualitative features [31]. to the contrary, in this paper we demonstrated a method for identifying user profiles using the interactions of the users as a single input on two datasets. We started by using a semantic mapping on top of the GOSPL database. For this mapping, we made an extension of the SIOC Core ontology. We use this ontology, since the concept of posts on an online platform is widely used, moreover the ontology is easy to reuse. The benefit of using a semantic mapping is that when we apply this mapping to multiple datasets (even from various platforms), we use a uniform way for requesting the different interactions.

After we obtained the quantitative number of interactions by each user, we pre-process the datasets. Since we deal with unequal means and variances we decided to standardize each dimension. Then we applied a PCA analysis in order to obtain the most interesting principal components for reducing the dimensionality of the data. In our datasets we originally used datasets of 10 and 11 dimensions. After reduction of the dimensionality, we ended up with a two-dimensional dataset in both datasets.

The method demonstrated the first dataset has five user profiles, where the second dataset only has three distinct user profiles. Though the second dataset covered a longer time frame in which the users of GOSPL could interact, we identified less distinct user profiles using the K-means clustering technique. In the second dataset, we did observe the differences were smaller between the found user profiles. This can partly be explained by the lower number of interactions made by users in the second dataset. Moreover the three active users (found in cluster 1) in this second dataset were responsible for 1,533 interactions, which is over 20% of the total interaction amount. Since we only look at the quantitative measurements of interactions as a basis, this method is very sensitive to distinguishing active users as a separate user profile.

In this proposed method we validate the cluster quality using silhouette coefficients. We choose this cluster validity technique since this technique provides a good measurement that combines measuring both internal homogeneity and external heterogeneity of clusters.

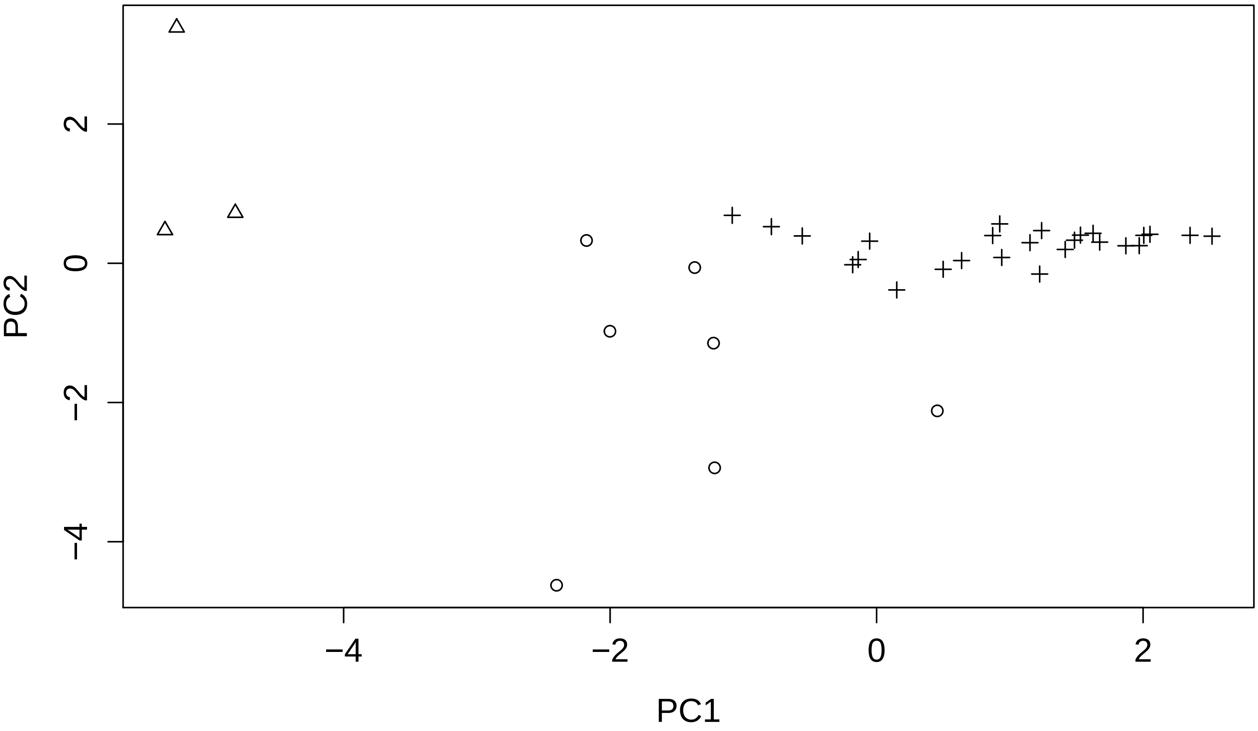

Fig. 6 Clusters of dataset II (clusters indicated with symbols)

Other validity techniques that could be used are *Dunn index*, *DaviesBouldin index*, and the *C-index* [15]. The silhouette coefficient is bounded between −1 for incorrect clustering and +1 for highly dense clustering. We are thus interested in clustering with this coefficient as high as possible.

We consider this paper as a first step in assigning roles automatically to the users based on the behavior of the user. The project leader of an ontology engineering project decides when to run this method, and after analysis of the found profiles he can rearrange his team of ontology engineers in order to work more efficient. This can either be by composing task groups of a similar user type or by composing new task groups by combining different user types.

Acknowledgements We would like to thank em. prof. dr. Meersman R. for his valuable input, and the statistical team prof. dr. Coomans D., prof. dr. Questier F. and Simons K. for giving input on clustering principles using PCA.

References

1. Armstrong RA, Slade SV, Eperjesi F (2000) An introduction to analysis of variance (ANOVA) with special reference to data from clinical experiments in optometry. Ophthalmic and Physiological Optics 20(3):235–241
2. Billsus D, Pazzani MJ (1999) A personal news agent that talks, learns and axplains. In: Proceedings of the 3rd Annual Conference on Autonomous Agents, AGENTS '99, pp 268–275
3. Cyganiak R, Bizer C, Garbers J, Maresch O, Becker C (2012) The D2RQ mapping language. http://d2rq.org/d2rq-language
4. Daga E, Gangemi A, Presutti V, Salvati A (2008) Ontology design patterns . org (odp). http://ontologydesignpatterns.org
5. Das S, Cyganiak R, Sundara S (2012) R2RML: RDB to RDF mapping language. W3C recommendation, W3C
6. De Leenheer P, Debruyne C, Peeters J (2009) Towards social performance indicators for community-based ontology evolution. In: Work-

shop on Collaborative Construction, Management and Linking of Structured Knowledge, Collocated with the 8th International Semantic Web Conference, ISWC '09

7. De Leenheer P, Christiaens S, Meersman R (2010) Business semantics management: A case study for competency-centric HRM. Computers in Industry 61(8):760–775
8. De Nicola A, Missikoff M, Navigli R (2009) A software engineering approach to ontology building. Information Systems Journal 34(2):258–275
9. Debruyne C, Nijs N (2013) Using a reputation framework to identify community leaders in ontology engineering. In: Meersman R, Panetto H, Dillon TS, Eder J, Bellahsene Z, Ritter N, De Leenheer P, Dou D (eds) Proceedings of On the Move to Meaningful Internet Systems: OTM 2013 Conferences - Confederated International Conferences: CoopIS, DOA-Trusted Cloud, and ODBASE 2013, OTM '13, pp 677–684
10. Debruyne C, Tran TK, Meersman R (2013) Grounding ontologies with social processes and natural language. Journal on Data Semantics 2(2-3):89–118
11. Falconer SM, Tudorache T, Noy NF (2011) An analysis of collaborative patterns in large-scale ontology development projects. In: Musen MA, Corcho Ó (eds) Proceedings of the 6th International Conference on Knowledge Capture, K-CAP '11, pp 25–32
12. Golder SA, Donath J (2004) Social roles in electronic communities. In: Proceedings of the 5th Conference of the Association of Internet Researchers, AoIR '04
13. Gruber T (1993) Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies 43:907–928
14. Guha R, Brickley D (2004) RDF vocabulary description language 1.0: RDF schema. W3C recommendation, W3C
15. Günter S, Bunke H (2003) Validation indices for graph clustering. Pattern Recognition Letters 24(8):1107–1113
16. Hautz J, Hutter K, Fuller J, Matzler K, Rieger M (2010) How to establish an online innovation community? the role of users and their innovative content. In: Proceedings of the 43rd Hawaii International Conference on System Sciences, HICSS '10, pp 1–11
17. Jolliffe I (1986) Principal Component Analysis. Springer Verlag, New York, NY, USA
18. Khatri V, Brown CV (2010) Designing data governance. Communicatons of the ACM 53(1):148–152
19. Kotis K, Vouros GA (2006) Human-centered ontology engineering: The HCOME methodology. Knowledge Information Systems 10(1):109–131
20. Lang K (1995) Newsweeder: Learning to filter netnews. In: Proceedings of the 12th International Conference on Machine Learning, ML '95, pp 331–339
21. Lieberman H (1995) Letizia: an agent that assists web browsing. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'95, pp 924–929
22. Lieberman H, Van Dyke NW, Vivacqua AS (1999) Let's browse: a collaborative web browsing agent. In: Proceedings of the 4th International Conference on Intelligent User Interfaces, IUI '99, pp 65–68
23. Linden G, Smith B, York J (2003) Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7(1):76–80
24. McDonald DW, Ackerman MS (2000) Expertise Recommender: a Flexible Recommendation System and Architecture. In: Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, CSCW '00, pp 231–240
25. Middleton SE, Shadbolt NR, De Roure DC (2004) Ontological user profiling in recommender systems. ACM Transactions on Information Systems 22(1):54–88
26. Nolker RD, Zhou L (2005) Social computing and weighting to identify member roles in online communities. In: Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence, WI '05, pp 87–93
27. Pinto HS, Staab S, Tempich C (2004) DILIGENT: Towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of ontologies. In: Proceedings of the 16th European Conference on Artificial Intelligence, ECAI '04, pp 393–397
28. Pontikakos C, Zakynthinos G, Tsiligiridis T (2005) Designing cscw system for integrated, web-based, cotton cultivation services. Operational Research 5(1):177–191
29. Prud'hommeaux E, Seaborne A (2008) SPARQL Query Language for RDF. W3C Recommendation, W3C
30. Resnick P, Iacovou N, Suchak M, Bergstrom P, Riedl J (1994) Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, CSCW '94, pp 175–186

31. Rowe M, Fernández M, Angeletou S, Alani H (2013) Community analysis through semantic rules and role composition derivation. Journal of Web Semantics 18(1):31–47
32. Schafer JB, Konstan JA, Riedl J (2001) E-commerce recommendation applications. Data Mining and Knowledge Discovery 5(1-2):115–153
33. Shardanand U, Maes P (1995) Social information filtering: Algorithms for automating word of mouth. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '95, pp 210–217
34. Tudorache T, Nyulas C, Noy NF, Musen MA (2013) Webprotégé: A collaborative ontology editor and knowledge acquisition tool for the web. Semantic Web Journal 4(1):89–99
35. Walk S, Singer P, Strohmaier M, Helic D, Noy NF, Musen MA (2015) How to apply markov chains for modeling sequential edit patterns in collaborative ontology-engineering projects. International Journal of Human-Computer Studies 84:51–66
36. Wang H, Tudorache T, Dou D, Noy NF, Musen MA (2013) Analysis of user editing patterns in ontology development projects. In: Meersman R, Panetto H, Dillon TS, Eder J, Bellahsene Z, Ritter N, De Leenheer P, Dou D (eds) Proceedings of On the Move to Meaningful Internet Systems: OTM 2013 Conferences - Confederated International Conferences: CoopIS, DOA-Trusted Cloud, and ODBASE 2013, OTM '13, pp 470–487
37. Wang H, Tudorache T, Dou D, Noy NF, Musen MA (2015) Analysis and prediction of user editing patterns in ontology development projects. Journal on Data Semantics 4(2):117–132

A Descriptive Statistics of Datasets

Descriptive statistics of the two datasets are provided in Tables 4 and 5. The dimensions of each table are:

- dim 1: interactions about glosses
- dim 2: interactions about lexons
- dim 3: interactions about constraints
- dim 4: interactions about supertype relations
- dim 5: interactions about equivalence between glosses
- dim 6: interactions about synonyms
- dim 7: interactions considering general requests
- dim 8: interactions considering replies
- dim 9: interactions to close topics
- dim 10: interactions about casting votes
- dim 11: interactions about semantic interoperability requirements

Table 4 Descriptive statistics of clusters in first dataset

		dim 1	dim 2	dim 3	dim 4	dim 5	dim 6	dim 7	dim 8	dim 9	dim 10
cluster 1	$\bar{x}$	25.67	28.50	16.67	9.50	0.33	9.33	0.67	13.50	69.33	118.33
(n = 6)	σ	10.03	12.23	9.67	9.71	0.82	6.02	1.63	12.55	19.63	91.93
	95% CI lower bound	15.14	15.67	6.52	-0.69	-0.52	3.01	-1.05	0.33	48.73	21.86
	95% CI upper bound	36.20	41.33	26.81	19.69	1.19	15.65	2.38	26.67	89.94	214.81
cluster 2	$\bar{x}$	14.00	11.40	9.30	0.20	0.60	1.40	1.20	35.60	34.70	172.40
(n = 10)	σ	8.30	3.69	14.13	0.63	0.97	2.12	1.23	14.86	17.52	61.03
	95% CI lower bound	8.06	8.76	-0.81	-0.25	-0.09	-0.12	0.32	24.97	22.17	128.74
	95% CI upper bound	19.94	14.04	19.41	0.65	1.29	2.92	2.08	46.23	47.23	216.06
cluster 3	$\bar{x}$	179.00	112.00	123.00	49.00	2.00	82.00	3.00	97.00	570.00	96.00
(n = 1)	σ	-	-	-	-	-	-	-	-	-	-
	95% CI lower bound	-	-	-	-	-	-	-	-	-	-
	95% CI upper bound	-	-	-	-	-	-	-	-	-	-
cluster 4	$\bar{x}$	32.00	36.33	29.00	6.67	0	10.67	2.00	50.00	108.00	240.00
(n = 3)	σ	11.79	15.95	19.29	6.11	0	3.06	2.65	23.07	54.62	136.61
	95% CI lower bound	2.71	-3.28	-18.91	-8.51	0	3.08	-4.57	-7.30	-27.68	-99.35
	95% CI upper bound	61.29	75.95	76.91	21.85	0	18.26	8.57	107.30	243.68	579.35
cluster 5	$\bar{x}$	7.27	5.73	2.14	0.36	0.09	0.82	0.09	7.18	10.59	47.05
(n = 22)	σ	9.16	6.42	3.98	0.95	0.43	1.59	0.29	9.68	12.56	40.16
	95% CI lower bound	3.21	2.88	0.37	-0.06	-0.10	0.11	-0.04	2.89	5.02	29.24
	95% CI upper bound	11.33	8.57	3.90	0.79	0.28	1.52	0.22	11.48	16.16	64.85

Table 5 Descriptive statistics of clusters in second dataset

		dim 1	dim 2	dim 3	dim 4	dim 5	dim 6	dim 7	dim 8	dim 9	dim 10	dim 11
cluster 1	$\bar{x}$	26.33	19.67	21.33	0	0	0	1.33	82.33	95.33	250.33	14.33
(n = 3)	σ	8.08	9.61	8.33	0	0	0	2.31	7.09	18.50	133.99	7.23
	95% CI lower bound	6.25	-4.20	0.65	0	0	0	-4.40	64.71	49.37	-82.53	-3.64
	95% CI upper bound	46.41	43.54	42.02	0	0	0	7.07	99.96	141.30	583.19	32.30
cluster 2	$\bar{x}$	6.57	20.71	12.00	0.43	0.57	5.71	0	41.71	38.57	132.29	10.14
(n = 7)	σ	5.97	16.49	11.25	0.79	0.98	6.68	0	22.93	16.21	64.19	7.88
	95% CI lower bound	1.05	5.46	1.59	-0.30	-0.33	-0.46	0	20.51	23.58	72.92	2.85
	95% CI upper bound	12.09	35.97	22.41	1.16	1.47	11.89	0	62.92	53.56	191.65	17.43
cluster 3	$\bar{x}$	3.08	6.85	3.62	0.42	0	0.23	0.04	14.31	16.35	94.65	3.27
(n = 26)	σ	4.07	6.87	5.73	1.39	0	0.65	0.20	13.40	19.21	56.31	3.77
	95% CI lower bound	1.43	4.07	1.30	-0.14	0	-0.03	-0.04	8.90	8.59	71.91	1.75
	95% CI upper bound	4.72	9.62	5.93	0.99	0	0.49	0.12	19.72	24.11	117.40	4.79