Semantic Formalization of Cross-site User Browsing Behavior

Julia Hoxha
Karlsruhe Institute of Technology
Institute of Applied Informatics and Formal Description Methods (AIFB)
Karlsruhe, Germany
julia.hoxha@kit.edu

Keywords
semantic log, cross-site browsing log formalization, classification of navigation logs, prediction with structured output, structured SVM

1. INTRODUCTION

Large amounts of data are being produced daily as detailed records of Web usage behavior, but the task of deriving knowledge from them still remains a challenge. Modeling and mining approaches are significant instruments to discover browsing patterns in such data and to understand how users browse Web sites.

There is an increasing body of literature on the investigation of clickstream data and navigation behavior modeling, with the majority focusing on data collected in a single site. Inspiring works [19] convincingly argue on the benefits of studying user behavior at multiple websites. Such approaches present significant potential to derive actionable behavioral knowledge and make better future forecasts, but they still have to tackle the problem of heterogeneity of the information encountered at different sites.

We approach the problem of usage data comprehensability at its root, addressing the issue of semantically formalizing cross-site user Web browsing behavior. Usage data (or usage logs) are syntactic representations of Uniform Resource Locator (URL) requests of pages and Web resources accessed by the site visitors. Due to the primarily syntactical nature of such requests, comprehension of users' browsing patterns is difficult. Hence, there is an urge for formalization approaches that leverage the semantics of the usage data in accordance with the domain they occurred.

As such, mapping usage logs to comprehensible events from the application domain helps to discover more insights about user behavior. While most approaches use flat taxonomies to represent such vocabulary, we deploy ontologies for structuring domain concepts and relations, since they ensure a richer semantic model of a Web site content.

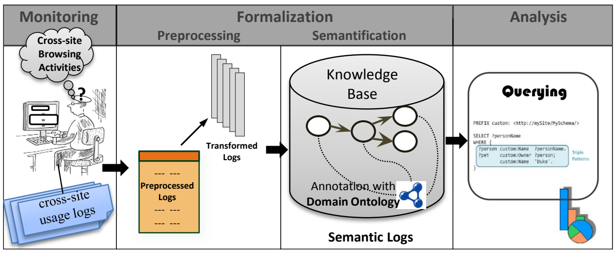

Figure 1: User Browsing Behavior Formalization Approach

This work aims at monitoring user behavior across multiple Web sites, logging clickthrough data upon agreement of Internet users. Each log entry is a tuple $\mathcal{L}$={*UserID, URL, timestamp*} of a user ID, URL of the accessed Web resource, and real time when this happened. These usage data logs are initially stored in raw form, as produced upon each user interaction.

The overall approach, illustrated in Fig. 1, comprises a series of steps, such as data preprocessing (human logs filtering, session construction, data transformation), formalization of usage logs, and techniques for their semantic enrichment.

This thesis will give the following contributions to the field:

- **Model for the formal and semantic representation of cross-site browsing logs**. I present the Web browsing Activity Model (WAM), expressed as an OWL-2-DL ontology, which enables a shared conceptualization of the knowledge from the various domains where the usage logs are recorded.
- **Techniques for the automatic extraction of the usage logs semantics** from heterogenous sources, in which the domain knowledge has a semi-structured or structured formal representation.
- **New approach for semi-supervised prediction with ontology-based output spaces**. This covers the problem of inferring the semantics of logs belonging to sites that do not offer a domain ontology. The contribution is a structured prediction algorithm formulated for the case of complex output objects rep-

resented as ontologies (with is-a hierarchies of classes and relations among them).

2. MOTIVATION

Existing approaches can benefit from leveraging usage data with semantics in the following ways: increase understandability of user behavior with respect to the application domain; enable analysis on higher levels of abstraction e.g. for parameters in URL (museum instead of Louvre), or for location (capital instead of Paris), which can be also used for privacy protection; allow formulation of more expressive queries for mining user behavioral patterns. Furthermore, enrichment of usage data with domain knowledge provides a broader context of user behavior, which can be exploited for more intelligent recommendation models.

The semantically-leveraged logs provide an added-value with respect to their syntactic representation in being useful inputs for techniques such as semantic pattern mining, next-step navigation prediction or user clustering, which usually assume that the semantics of logs exists or are manually derived. A more beneficial aspect is the extension of these techniques to deal with cross-site browsing data and not only a single Web site.

2.1 Applications

An interesting application is the integration of domain knowledge in the process of discovering usage patterns. This helps to increase the precision, and hence interpretation of the retrieved patterns, while ensuring different level of abstraction. I present two approaches for discoverying browsing behavior patterns, while using as basis the formal and semantic representation of logs:

I.Ontology-based Web usage mining
The first application deals with the automatic mining of frequent patterns from the sequence of event logs, which are enriched with description from domain ontologies. While recent trends in Web Usage Mining (WUM) have put the emphasis on the exploitation of ontologies to the pattern mining process, yet they share two limitations: the ontologies are either restricted to representations of class taxonomies while ignoring relates among the concepts, which reduces the problem back to the traditional generalized sequential pattern discovery [?], or they are restricted to a single ontology (single Site) that is assumed to be completed with relations. Because of the hetegoreneity of Web sites and respective domain knowledge, our setting requires a mining technique that addresses the problem when there are multiple ontologies in background and not all the relations among the concepts are established. Hence, they still need to be inferred during the mining process.

As a contribution to the WUM field, with practical motivation from the Web personalization field, I propose an approach for mining frequent sequential patterns in the presence of multiple domain ontologies. The mined patterns can, then, easily be extended to association rules [?], which provides predictions for the user's next step navigation preference.

II. Pattern discovery with $\mathcal{DL}$-LTL expressive queries
In this application, patterns are discovered from the corpus of the semantically formalized logs upon issuing specific queries that express semantic and temporal conditions of usage behavior.

While the first application (mining) concentrates on the semantics of the logs, another crucial aspect to consider when analyzing browsing behavior is also its temporal dynamic. Additional aspects of user browsing behavior can be discovered if reasoning not only with semantic constraints, but also with more expressive temporal conditions is made possible. I introduce an approach to formulate queries using a temporalized description logic called $\mathcal{DL}$-LTL, which combines $\mathcal{SROIQ}$ [9] with Lineal Temporal Logic (LTL) [1] over finite traces.

It is further shown how to search for behavioral patterns from the usage logs applying a query answering technique, which is based on current model checking tools. This allows to automatically retrieve sessions of user browsing events that satisfy a set of semantic and temporal conditions. The adaptation and application of the $\mathcal{DL}$-LTL logic and these techniques for the setting of Web usage analysis are novel.

3. STATE OF THE ART

The contributions related to this thesis are grouped into works dealing with 1) the modeling of user browsing behavior at multiple Web sites, 2) formal and semantic description of usage logs, and 3) exploitation of ontologies in Web usage mining, and 4) prediction of structured data.

3.1 Modeling Cross-site User Browsing Behavior

Interest to characterize online behavior has started much earlier with works such as those of Catledge *et al.* [5], and Montgomery *et al.* [17] that try to identify browsing strategies and patterns in the web. Browsing activity has been studied and modeled, e.g. Bucklin *et al.* [4] and others, usually exploiting server-side logs of visitors in a specific website.

Regarding the modeling of browsing behavior at multiple websites, Downey *et al.* [8] propose a state machine representation for describing search activities. The present an approach for modeling and analyzing user behavior, focusing on the search activities and what users do when they depart the search engine. Park and Fader [19] present a stochastic timing model of cross-site user visit behavior, using information from one site to explain the behavior at another. While, Johnson*et al.*[12] study online search and browsing behavior across competing e-commerce sites.

The works in this category do not particularly apply semantic techniques or ontologies for behavior modeling.

3.2 Ontologies in Usage Mining

There is an extensive body of work dealing with usage log analysis and mining, but we focus on the combination of these techniques with semantic technologies, which start with contributions such as Stumme *et al.* [21] and Oberle *et al.* [18]. In this field, research has been mostly focused on search query logs or user profiling. Recent approaches, which use semantics for extracting behavior patterns from web navigation logs, are presented by Yilmaz *et al.* [25] and Mabroukeh *et al.* [14].

Vanzin *et al.* [24] present ontology-based filtering mechanisms for the retrieval of Web usage patterns. More recently, Mehdi *et al.* [15] tackle the problem of mining meaningful usage patterns and exploit the impact of ontologies to solve this problem. These works are restricted to only one domain and not cross-site browsing behavior. Hence, they mostly deal with a mining problem in the presence of a single ontology. It is interesting to explore further the discovery of patterns when multiple domain ontologies are involved, considering the establishment of mappings between them as an additional requirement of the mining process.

It is important to note though, that the process of enriching of logs with semantics is not the central problem of these works. They mostly use the ontological knowledge in the background for leveraging or optimizing the mining techniques.

3.3 Semantic Formalization of Usage Logs

This group consists of works that directly deal with semantic annotation of usage logs, hence mapping the requests of Web resources to meaningful concepts from the application domain. **d'Aquin *et al.*[7]** present the UCIAD platform[1], which applies annotation of user-centric activity data. It relies on pre-defined URL patterns to characterize accessed resources over which the activities are realised, and therefore their respective semantics. As part of setting up the platform, it is initially defined which is the set of websites that are present on the considered server, as well as the URL patterns,expressed as regular expressions, enable to recognise webpages as parts of these websites. Similarly, definitions of the user activities are also manually made in the setup process, in order to characterize and give semantics to the user actions.

The work of **Tvarozek *et al.* [23]**, while actually focusing on an architecture for the personalized presentation layer of Web-based information systems, covers in one of its techniques the problem of semantically annotating usage logs. In order to create comprehensive logs of user actions, the logs browsing events captured by a client side monitoring tool, as well as server-side logging data, are enhanced with semantics from the Web sites content using a SemanticLog tool. This tool is based on a semantically-enabled portal, which means that there is a conceptual ontology in the background of the site. The mapping of an interaction of the user with parts of the Web site, then use this ontology to generate the annotation of the user action. In this case, the semantics of the logs are not inferred, but rather defined in background as part of the engineering of the site. Still, this can be feasible only in the case when one is in charge of the content of the site, and also restricted to a small set of sites. Additional manual effort in the engineering process is needed to generate the semantic annotations.

Stühmer *et al.*[20] focus on processing complex events of user interactions with annotated Web pages, and they also present an approach for capturing and lifting these events in RDF. Hence, instead of dealing with the syntactical form of events, they also address leveraging logs with semantic information, which pertains to the actual domain knowledge of the Web page. As in the previous work, this technique also assumes the presence of a semantically-enabled Web site. In this case, RDFa is used to support the semantics embedded within actual Web page data and allow reusable semantic markup inside of Web pages.

The related works in this group are restricted to a manual approach for enriching the logs with semantics. This limitation poses a significant burden when we need to analyse browsing behavior at various Web sites, which leads to immense efforts of extracting the semantics of logs and mapping them to respective domain ontologies. Moreover, it is assumed that the domain ontology is provided. This leaves the problem of inferring (learning) the semantic types of logs for non-semantically enabled sites still a challenge.

3.4 Prediction of Structured Data

Machine Learning today offers a broad range of methods for classification and regression, but only a few cover the problem of predicting complex objects, such as trees or graphs. The approaches dealing with prediction of structured and interdependent output data are principally grouped into those using probabilistic models (e.g. Conditional Graphical Models, HMM) and those using discriminative models (e.g. Max-Margin Structured Classification [**?**], Energy-Based Models [**?**], SVM).

In the latter group, Support Vector Machines (SVM) for structured and interdependent output spaces [22, 11] offer solid theoretical foundations, as well as very high efficiency for the structured prediction approach. While structural SVMs provide a generalized formulation of the learning problem, its state of the art applications cover only the case when the output object are sequences or trees.

There is still the need to reformulate the learning problem, and further adapt the SVMs for the case when the output instances are objects represented as ontologies. In this case, ontologies comprise not only a hierarchical structure of the classes (is-a hierarchy) in the output space, but also a set of semantic relations between these classes. The difficulty of the prediction problem now increases, since it requires learning a model that takes into account the semantics of the ontology in the output space, which is an additional requirement when compared to the current techniques that deal with general graphs or trees.

4. FORMALIZATION OF WEB BROWSING BEHAVIOR

When browsing the Web, users interact with Web resources via browsers interface (e.g. clicking links, submitting HTML forms, etc.). These interactions can be recorded as usage data in forms of Web server or client-side logs. We use the term *browsing event* to describe the basic component of user behavior in performing activities with the Web browser directly.

EXAMPLE 1. **(Cross-site Browsing Logs)**
ID Time User Action

[1]http://uciad.info/ub/

1 [17:04:14:35] http://www.avis.com/car-rental/reservation/ start-reservation.ac?resForm.pickUpLocation=Lyon
1 [17:11:49:21] http://www.google.de/search?q=Lyon+www2012
1 [17:11:49:33] http://dbpedia.org/page/Lyon
1 [17:11:49:39] http://data.semanticweb.org/conference/ www/2011/demo/a-demo-search-engine-for-products

In this running example of usage logs, the user starts a car rental reservation at Avis, next performs search at Google, and then visits sequentially the page of Lyon at DBpedia [2] *and the page of a demo paper at Semantic Web Dog Food* [3]*. The last two sites are semantically-enabled, thus, offer a domain ontology and data publishing as Linked Open Data.*

If each log entry is represented as a meaningful event from the application domain where it is issued, then user behavior becomes more comprehensible. The context of the event can be extended with additional information from the domain (as explained in Sec. 4.2). □

We aim at monitoring user behavior across multiple Web sites, logging clickthrough data upon agreement of Internet users. Each log entry is a tuple $\mathcal{L} = \{UserID, URL, timestamp\}$ of a user ID, URL of the accessed Web resource, and real time when this happened. These usage data logs are initially stored in raw form, as produced upon each user interaction.

Our formalization approach, illustrated in Fig. 1, comprises a set of techniques, such as data preprocessing, human logs filtering, session construction, formal description of logs, and semantic enrichment. In the following sections, we present the definitions upon which this work is based. We then focus on the semantic enrichment approach, which extends our previous work [10].

The contributions of this work are: 1) a model to formally and semantically structure usage logs, 2) an approach for the automatic extraction of the semantics of usage logs from heterogenous sources, in which the domain knowledge has a semi-structured or structured formal representation, and 3) a new approach for semi-supervised prediction with ontology-based output spaces

4.1 Formal Model for the Representation of Logs

Definition 1. **(Event)** *We define a browsing event as a tuple $e = (l, \mathcal{T}, P, t)$, where l is the full URL invoked, $\mathcal{T}$ is a set of event types for which this event qualifies, $P=\{p_1, ..., p_l\}$ is a set of parameters and t is the* occurrence time. *For simplicity, we denote event time by $e_i.t$ and set of event types by $e_i.\mathcal{T}$.*

Each user browsing activity recorded in logs is physically represented by a URL, but conceptually it comprises an event that serves a particular function and relates to a specific content. We give meaning to each event issued as an HTTP request in the logs, by mapping its respective URL to domain concepts according to two dimensions: content and function. An event resulting from the interaction of a user with a specific Web page serves a particular function (e.g. searching, browsing, login, etc.) related to some content (e.g. flight reservation, organization, hotel, etc.).

Definition 2. **(Event Type)** *An event can be mapped to several types denoted by the set $\mathcal{T} = \{\mathcal{T}_c, \mathcal{T}_f\}$, where $\mathcal{T}_c$ is the type of content to which this event relates, $\mathcal{T}_f$ is the type of function this event serves.*

We have extended the definition of a browsing event with parameters, which can be extracted based on the information contained in the URL l. We consider three main conceptual elements in a link: URL base, variable names, and values. Based on the typical convention of URL formation, we syntactically split the link into two basic parts: URL base, which defines domain name, and the rest of the URL is used to extract input variables, which are modeled as event parameters.

Definition 3. **(Parameter)** *An event parameter p, which can be further classified as input or output parameter, is a pair p=(v_{name}, v_{value}) consisting of variable name and value.*

Events are grouped into sessions, which represent a period of sustained Web usage. The boundaries of a session are normally determined by temporal and behavioral factors (e.g. browsing intention). We follow previous research [4] in deploying an heuristic, which starts a new session after an idle period of *30 minutes* between the browsing events.

Definition 4. **(Session)** *We denote a user session as a tuple $S = \{s, T_s, T_e, U\}$, where $s = \langle e_1, e_2, ..., e_n\rangle$ is an ordered sequence of browsing events performed from user U, such that $e_i.t \leq e_{i+1}.t$ for all i, where i denotes the event order in the sequence. Furthermore, T_s is the starting time and T_e the ending time of the session, such that $T_s \leq e_i.t \leq T_e$.*

Ordering of events in a session is used later as a feature for the supervised learning of event types, when they are not available in the domain ontology.

For the realization of these concepts, I have use a Web Browsing Activity Model (WAM), which I formalize as an ontology (Fig. 2). This is also presented in the paper Hoxha et al. [10]

Classes and Properties. Classes in WAM are divided in three groups: Core classes, External classes, and Type classes. External classes are basic concepts that I reuse from well-established ontologies. Each `wam:Event` is a subclass of the concept `event:Event` from the *Event ontology* [4].

[2] http://dbpedia.org
[3] http://data.semanticweb.org
[4] http://purl.org/NET/c4dm/event.owl#

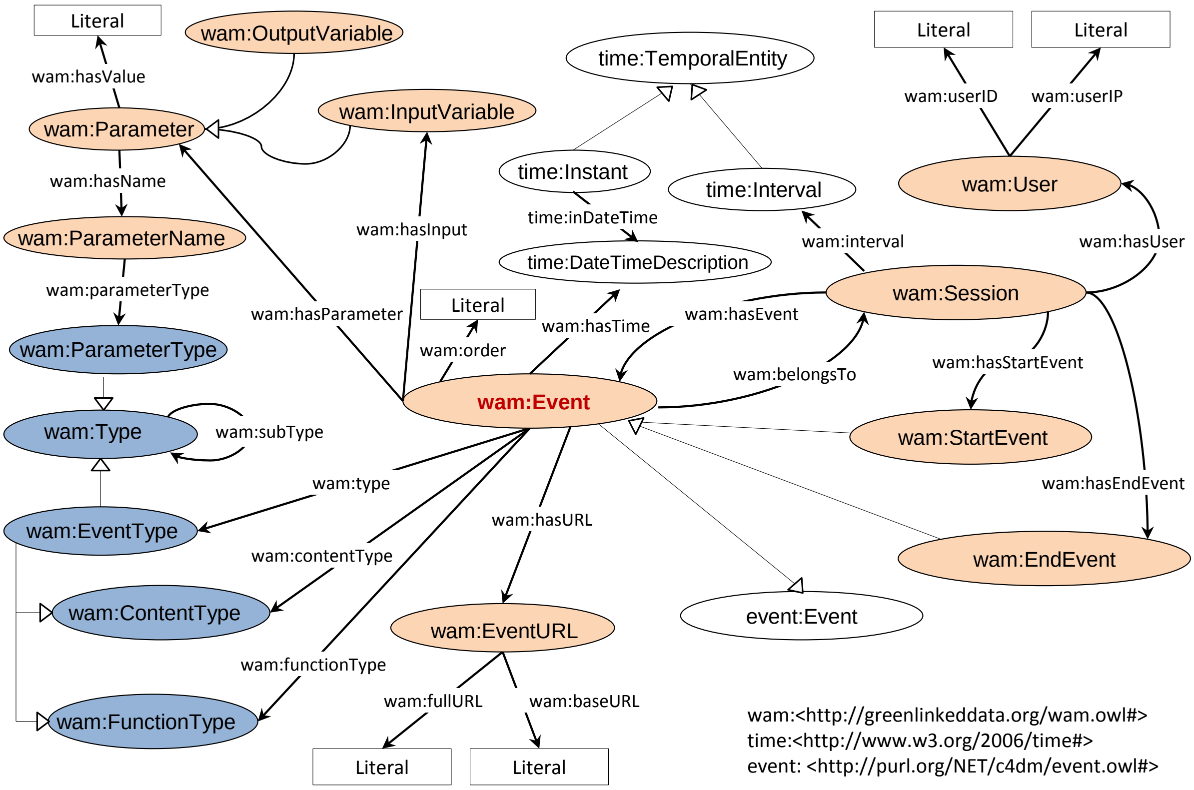

Figure 2: WAM Ontology

Each `wam:Session` has one `wam:StartEvent` and one `wam:EndEvent`, both of type `wam:Event`. Class `wam:User` is simply characterized by user IP address and ID, but the ontology allows flexible future extendability with user profiles or other attributes (e.g IP-based geographical location). To annotate of event timestamps and session interval, I reuse basic concepts from *OWL Time ontology*, [5] which models knowledge about time such as temporal units, instants, etc.

The ontology is expressed in OWL-2-DL with underlying $\mathcal{SROIQ}$ logic [9].

4.2 Semantic Enrichment using Domain Knowledge

The main focus of the semantic enrichment approach, formally described in algorithm Alg. 1, is to find the content types belonging to each event [6]. For each URL resource request l in the logs, we use the Content Negotiation mechanism[7] to retrieve the respective RDF representation, if it is available. Based on the RDF, we identify the Uniform Resource Identifier (URI) of the resource (Alg.1, line 4). We also retrieve the domain ontology O_d of the respective Web domain, as well as the class to which the given resource belongs (querying on *rdf:type*) (line 6). A resource may be a member of many classes in the O_d, therefore we consider all of them as instances of content type (line 9).

[5]http://www.w3.org/2006/time#
[6]In this work, we focus on the contentType class only
[7]http://www.w3.org/Protocols/rfc2616/rfc2616-sec12.html

Algorithm 1 Automatic semantic enrichment of a browsing event: *findContentTypes(s)*

Require: Ordered sequence of events $s = \langle e_1, e_2, ..., e_n \rangle$
Ensure: Update knowledge base with new ABox assertions α_t related to content types
  for all $e_i \in s$ **do**
    Get the URL $e_i.l$ **of the event**
3:     Retrieve RDF_l=*retrieveRDF*($e_i.l$) the RDF/XML representation
    Get resource URI R_l=*identifyResourceURI*(RDF_l)
    $\mathcal{O}_b = getDomainOnt(RDF_l)$;
6:     Find $\mathcal{T}_c = classMembership(R_l)$
    for all $T \in \mathcal{T}_c$ **do**
      Abox Assertions
9:       $\alpha_t = \{ContentType(T), contentType(\mathtt{e}_i, T)\}$
    end for
  end for
12: **Serialize assertions** α_t **and update knowledge base**

An example of enriching an event with semantics is illustrated in Figure 3. This is the last log entry of our running example in Ex. 1. Initially, the raw log is just a syntactic representation of the URL request (in this case a demo paper). We retrieve the respective RDF representation and identify the resource with its URI. Querying (via SPARQL[8]) the domain ontology, here the *SWRC* publications ontology, we can enrich the eventt's semantics with additional knowledge.

[8]http://www.w3.org/TR/rdf-sparql-query/

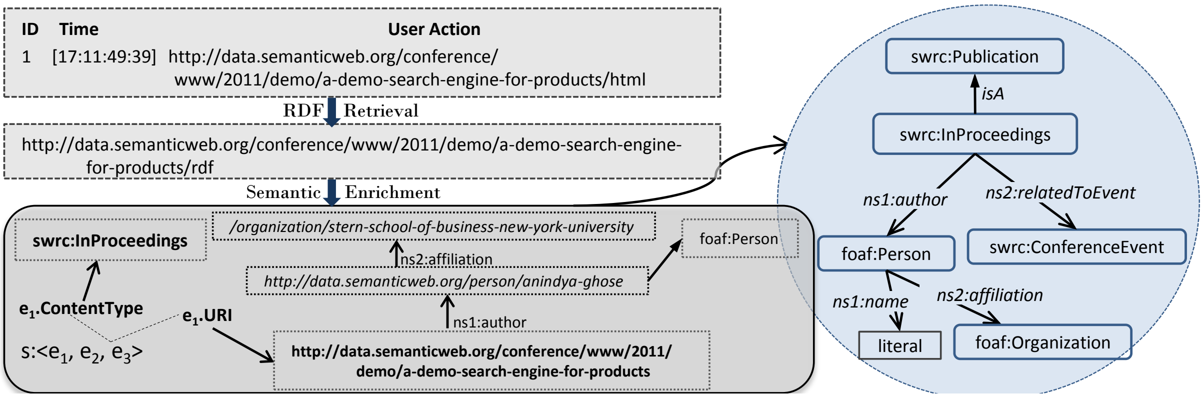

Figure 3: Semantic Enrichment of Usage Logs

We find that this resource is a *Demo* of type *InProceedings*. We can further extend the context with information like the *conference* WWW2011 it belongs, the conference *location*, the *author* of the paper, the author's *affiliation*, etc.).

5. SUPERVISED LEARNING OF EVENT CONTENT TYPES

5.1 Problem Definition

Finding the `contentType` class of a browsing event can be formulated as a classification problem, borrowing from the field of machine learning. After the deployment of the formalization and automatic semantic enrichment approach of Sec. 4, we generate a session $s_k = \langle e_1, e_2, ..., e_n \rangle$ (Def. 4) as a sequence of semantically-annotated browsing events (as in Def. 1). For most of the events in the sequence, we are able to automatically find and assign a `contentType` class from the domain ontology. But, there are also two events in s_k, where no `contentType` class could be derived. Hence, we follow a second step of semantic enrichment that comprises a supervised technique for learning the class, based on the observed examples (i.e. already formalized events in the overall sessions). We need to assign a particular event from the logs to a predefined class, being in our case the `contentType` of this event. Hence, we approach our problem as a classification task, which learns a function $f : E \rightarrow \mathcal{C}$ that maps an event $e_i \in E$ s.t. $e_i = (l_i, \mathcal{T}_i, P_i, t_i)$ (as in Def. 1) to an output class $c_i \in \mathcal{C}$. In our case, $\mathcal{C}$ is a set of classes belonging to an ontology $\mathcal{O}$.

5.2 Classification with Structural Support Vector Machines

In our approach, we use the generalized formulation of multiclass SVM learning [22]. We are interested on the problem of learning a function $f : \mathcal{X} \rightarrow \mathcal{Y}$, which maps input instances $\mathbf{x} \in \mathcal{X}$, which in our setting consist of the events in the logs, to discrete outputs $\mathbf{y} \in \mathcal{Y}$ that consist of arbitrarily numbered labels representing `contentType` classes in our events ontology.

Lets consider the case of finding a function f that maps each event $\mathbf{x}_i$ from usage logs to one of the classes in $\mathcal{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_n\}$. The problem addressed is to learn a discriminant function $F : \mathcal{X} \times \mathcal{Y} \rightarrow \Re$ over input/output pairs, so that for a given input $\mathbf{x}$, we can derive a prediction by maximizing F over the response variables:

$$F(\mathbf{x}; \mathbf{w}) = \underset{\mathbf{y} \in \mathcal{Y}}{argmax}\ F(\mathbf{x}, \mathbf{y}; \mathbf{w})) \qquad (1)$$

In our case, we deal with a multi-class classification problem [6], where $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_K\}$ is the input set of events of sessions from the usage logs, $\mathcal{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_N\}$ is the set of output classes from ontology $\mathcal{O}$, and $\mathbf{w} = (w_1, ..., w_N)$ is a stack of vectors with w_n being a weight vector for the class $\mathbf{y}_n$. We use the following formulations of the linear discriminant functions F:

$$F(\mathbf{x}, \mathbf{y}_n; \mathbf{w}) = \langle \mathbf{w}_n, \Phi(\mathbf{x}) \rangle \qquad (2)$$

where $\Phi(\mathbf{x}) \in \Re$ is the vector of numeric features extracted from $\mathbf{x}$. SVM, then, solves the following optimization problem:

$$\underset{\mathbf{x}, \xi}{min} \frac{1}{2} \sum_{i=1}^{N} \| \mathbf{w}_i \|^2 + \frac{C}{K} \sum_{i=1}^{K} \xi_i \qquad (3a)$$

$$\forall i, \forall \mathbf{y} \in \mathcal{Y} \setminus \mathbf{y}_i : \langle \mathbf{w}, \Phi(x_i) \rangle \geq 100\Delta(\mathbf{y}_n, \hat{\mathbf{y}}_n) - \xi_i \qquad (3b)$$

with regularization parameter C and slack variables ξ_i for margin violations (for details see [11]). The learning algorithm optimizes the error rate during training, minimizing prediction loss defined by a function $\Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \Re^D$, where $\Delta(\mathbf{y}_n, \hat{\mathbf{y}}_n)$ is the loss of predicting $\mathbf{y}_n$ when the correct output is $\hat{\mathbf{y}}_n$.

Reasons for choosing structural SVM

There are several reasons for choosing structural SVM as our classification approach. Firstly, SVMs in general are shown to perform better in building complex and accurate models [11], particularly in settings similar to ours such as Web page categorization or purely URL-based page classification [13, 2]. Secondly, SVMs deal very well with sparse and highly dimensional data, as is the case of the huge and heterogeneous amounts of cross-site usage logs, which lead to feature vectors that are large and highly sparse. At last, structural SVMs enable learning for complex and interdependent objects of the output space, leading us towards an extension of our approach in learning a formal, structured ontology with class relationships for the classification of events (i.e. requested resources) in the usage logs.

For our classification task, we follow a procedure comprising the following steps:

Preparation of Training and Testing Datasets. After the logs have been semantically formalized using our formalization approach, we select a portion of the data for the classification problem. Initially, since the formalized logs are represented as RDF triples and stored in a repository, using SPARQL queries we extract two sets of data for training and testing, each of them containing a huge vector of session id, URL of event and order of event belonging to that session. We then prepare training and test datasets, respectively. Since supervised learning needs labeled data, a part of those generated from the mapping to the domain ontology, which serve as ground truth values. The labels[9] that are not found in the ontology are annotated manually.

Feature Selection. We select different categories of features for the classification of event types. We first experiment with whole tokens (no stemming is performed) of URLs, and with the letter n-grams of the tokens [13]. We also test *sequential features*, such as sequences of pairs of tokens in the URL, referred as the *precedence bigrams*. We further propose a new feature based not only on the URL of the event, but on the sequential information related to the session in which the event belong (*sequential neighbors.*). In this case, the tokens of the neighboring events are also included as features.

Feature Vector Representation. As explained earlier (Sec. 5.1), SVMs require that each instance in the input space is represented as a vector of real numbers. Hence, we convert our inputs into vectors of numeric values. In order to construct such feature vectors, we follow a series of preprocessing steps aligned with our definition of the features. Preprocessing includes tokenization, n-gram generation, and precedence bigram formation. Tokens or ngrams derived from the URL of the event serve as binary features.

Model Selection. We experiment with the linear kernel of structural SVM, motivated by the following reasons: high dimensionality of the feature vectors, huge number of features, and high number of classes/labels. We experiment with different values of the regularization parameter C.

Evaluation Measures. To evaluate the performance of our classification approach, we use the F-measure metric, which is the harmonic mean of precision (π) and recall(ρ), defined as follows:

$$\pi_i = \frac{TP_i}{TP_i + FP_i}, \rho_i = \frac{TP_i}{TP_i + FN_i} \tag{4}$$

$$F_i = \frac{2\pi_i\rho_i}{\pi_i + \rho_i}, macroF_1 = \frac{\sum_{i=1}^{N} F_i}{N} \tag{5}$$

where TP_i (True Positives) is the number of instances assigned correctly to class i; FP_i (False Positives) is the number of instances that do not belong to class i, but are assigned to class i incorrectly; and FN_i (False Negatives) is the number of instances not assigned to class i, but which actually belong to this class.

[9]Terms *label* and *class*, as well as *instance* and *event* are used interchangeably.

The F-measure values are in the interval (0,1), and larger values correspond to higher classification quality. To compute the overall F-measure score of our multi-class classification problem, we use *macro-averaging* (Equation 5) as a binary evaluation measure across the overall N classes.

We have conducted experiments with datasets of real-world usage logs. In section 6 we provide details on the characteristics of the datasets used for training and testing, as well as report on the evaluation results of these experiments.

6. EVALUATION

6.1 Formalization Approach

We provide a Java SE implementation of the introduced formalization approach, deploying the steps of processing usage logs, cleaning, and formalization with WAM ontology (whose consistency is checked with Pellet 1.5.2 reasoner). We have further implemented the step of automatic semantic enrichment of events, for which we read and query using Jena Framework [10].

In order to show the feasibility of this approach, we performed experiments by semantically formalizing logs from the USEWOD datasets [3] featured in Table 1. The formalized sessions and events are serialized in RDF, and then imported via OpenRDF Sesame Core 2.6.0 API[11] into a repository of a Sesame Framework [12] that is made available online [13]. Overall, we processed nearly one month of usage logs from large Web sites (such as DBPedia), proving that the approach is scalable and able to retrieve the content types classes of more than 80% of events.

Regarding the practicality of the proposed approach, which requires the existence of semantically-enabled websites or sites that include RDF annotations, we note that the percentage of such sites in the Web is now continuously and quickly increasing (for details see [16]).

6.2 Supervised Learning Approach

Experimental Setup. For our supervised learning experiments, we used two datasets D_1 and D_2 of different sizes extracted from the repository of events generated in the first step of our formalization. These are the events belonging to the two weeks of the SWDF part of Table 1. For both datasets we prepared training and testing sets. The test sets contain events for which the content type was not automatically found. We report on the characteristics of these datasets in Table 2.

For dataset D_1, we chose usage logs of two random consecutive days, extracting the events of one day (3. July) for the training and events of another day (2. July) for testing. Whereas for D_2, we chose a larger set comprising the logs of all the days from both weeks.

We use the implementation $structSVM$[14] of structural SVMs

[10]http://incubator.apache.org/jena/
[11]http://www.openrdf.org/doc/sesame2/api/
[12]http://www.openrdf.org/
[13]http://46.4.66.131:8080/openrdf-workbench/repositories/wam/query
[14]http://svmlight.joachims.org/svm_struct.html

Table 1: Results of the Formalization Approach

	SWDF	DBPedia 3-3
Monitoring Period	01/07/09-13/07/09	01/07/09-13/07/09
Nr. Sessions	2831	31893
Average nr. sessions/day	235.92	2899
Nr. Triples	277788	> 3million
Nr. Events	10437	426k
Mode nr. events/session	4	10
% events with contentType classes	83%	81%

Table 2: Description of training and testing datasets

	Dataset D_1		Dataset D_2	
	Training set	**Test set**	**Training set**	**Test set**
Nr. events/set	974	1152	4676	4957
Total nr. events	2126		9632	
Nr. classes	66		82	

with the multi-class formulation. After experimenting with different values of the regularization parameter C, we chose the value 5000 to be the best one. For training, we follow a three-fold cross-validation approach.

Experimental Results. In our experiments, we use the token feature as the baseline. All the other experiments additionally use each of the other features. We report on the zero/one-error (%) and macro-F1 measures of our results.

As can be observed in Table 3, in particular the ngram (N) and sequential neighbor features (S) play an important role in increasing the classification accuracy. For D_1 we see that the combination of features N and S yields the most optimal results, since the error is the smallest, while still keeping a high value of the macro-F_1 measure (which is a harmony of precision and recall averaged accross all classes)[15]. For D_2 we note that the precedence bigram feature gives the best classification in terms of the 0/1 error rate. Still, as in D_1, the impact of the sequential neighbor feature yields the best combination of both the error and the overall averaged F_1 score. This proves our expectation that users sequentially browse related resources, which can help us derive missing semantics.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we address the issues of modeling user Web browsing behavior, tackling the problems related to information heterogeneity by using semantics. We present an approach for the formalization of such behavior across multiple sites based on a newly-introduced Web browsing Activity Model (WAM). A crucial part of the formalization is a two-staged semantic enrichment of logs, which maps them to events with comprehensible content types from the application domain. In order to find such semantic annotations of the logs, in the first stage we perfom an automatic technique to retrieve the semantic types of logs from existing domain ontologies of Web sites. To annotate the remaining logs of those sites that do not provide a formal domain ontology, we deploy a supervised learning technique via a multi-class classification formulation. We explore for the first time the use of Support Vector Machines with structural and interdependent output spaces, as well as the exploration of new session-related sequential features for the semantic classification of usage logs.

The semantically-leveraged logs provide an added-value with respect to their syntactic representation in various ways: allow for more expressive formulation of queries to discover user navigation patterns; are useful input for techniques, such as semantic pattern mining, next-step navigation prediction or user clustering, which usually assume that these semantics of logs exists or are manually derived. A more beneficial aspect is the extension of these techniques to deal with cross-site browsing data and not only single Web sites.

We have implemented the overall formalization approach with both stages of the semantic enrichment and performed experiments with real-world datasets of usage logs. We show that the extension with the supervised classification technique increases considerably the annotation accuracy. The introduced sequential features play an important role in ensuring a higher classification quality.

We plan to further investigate the techniques of learning semantic types (in particular function types) of usage logs, especially for semi-supervised techniques that reduce the effort of manually labeling training data. More interestingly, this work lays the foundations for a promising learning problem where the output space is not a set of classes, but a structured, formal ontology containing also the relations among concepts. We will elaborate on these aspects in our future work.

8. REFERENCES

[1] C. Baier and J.-P. Katoen. *Principles of Model Checking.* The MIT Press, 2008.

[15] The experiments on D_2, whose results are reported as N/A , were not supported by our machine because of the high dimesionality of feature vectors.

Table 3: Macro-F1 measure of the experimental results (Regularization parameter $C = 5000$)

	Dataset D_1			Dataset D_2		
Feature Category	Nr. Features	zero/one-error (in %)	Macro-F1	Nr. Features	zero/one-error (in %)	Macro-F1
Token (T)	1357	14.40	0.79	4341	13.04	**0.75**
Trigram (N)	4673	14.41	**0.84**	11205	12.99	0.69
Precedence Bigram (P)	3385	14.75	0.82	11060	**11.80**	0.58
Sequential Neighbors (S)	4071	13.54	0.67	13023	**12.02**	**0.63**
S+P	6099	14.06	0.73	15647	N/A	N/A
N+S	7387	**13.45**	**0.74**	19887	N/A	N/A

[2] E. Baykan, M. Henzinger, L. Marian, and I. Weber. A comprehensive study of features and algorithms for url-based topic classification. *TWEB*, 5(3):15, 2011.

[3] B. Berendt, L. Hollink, V. Hollink, M. Luczak-Rösch, K. H. Möller, and D. Vallet, editors. *USEWOD2012-2nd International Workshop on Usage Analysis and The Web of Data*, Lecture Notes in Computer Science. Springer, 2012.

[4] R. E. Bucklin and C. Sismeiro. A Model of Web Site Browsing Behavior Estimated on Clickstream Data. *Journal of Marketing Research*, XL:249–267, Aug. 2003.

[5] L. D. Catledge and J. E. Pitkow. Characterizing browsing strategies in the world-wide web. In *Computer Networks and ISDN Systems*, pages 1065–1073, 1995.

[6] K. Crammer, Y. Singer, N. Cristianini, J. Shawe-taylor, and B. Williamson. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:2001, 2001.

[7] M. dÁquin, S. E. L., and E. Motta. Semantic technologies to support the user-centric analysis of activity data. In *Workshop on Social Data on the Web Workshop (SDoW) at ISWC*, 2011.

[8] D. Downey, S. T. Dumais, and E. Horvitz. Models of searching and browsing: Languages, studies, and application. In *Proceedings of IJCAI*, pages 2740–2747, 2007.

[9] I. Horrocks, O. Kutz, and U. Sattler. The even more irresistible $\mathcal{SROIQ}$. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR2006)*, pages 57–67, June 2006.

[10] J. Hoxha, M. Junghans, and S. Agarwal. Enabling semantic analysis of user browsing patterns in the web of data. In *USEWOD Workshop at the 21st International World Wide Web Conference (WWW2012)*, volume abs/1204.2713, 2012.

[11] T. Joachims, T. Hofmann, Y. Yue, and C.-N. J. Yu. Predicting structured objects with support vector machines. *Commun. ACM*, 52(11):97–104, 2009.

[12] E. J. Johnson, W. W. Moe, P. S. Fader, S. Bellman, and G. L. Lohse. On the depth and dynamics of online search behavior. *Manage. Sci.*, 50:299–308, March 2004.

[13] M.-Y. Kan and H. O. N. Thi. Fast webpage classification using url features. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, CIKM '05, pages 325–326, New York, NY, USA, 2005. ACM.

[14] N. R. Mabroukeh and C. I. Ezeife. Using domain ontology for semantic web usage mining and next page prediction. In *CIKM*, pages 1677–1680, 2009.

[15] A. Mehdi, P. Valtchev, R. Missaoui, and C. Djeraba. A framework for mining meaningful usage patterns within a semantically enhanced web portal. In B. C. Desai, C. K.-S. Leung, and S. P. Mudur, editors, *C3S2E*, ACM International Conference Proceeding Series, pages 138–147, 2010.

[16] P. Mika and H. Potter. Metadata statistics for a large web corpus. In *Proceedings of the Linked Data Workshop (LDOW) at the International World Wide Web Conference*, 2012.

[17] A. L. Montgomery and C. Faloutsos. Identifying web browsing trends and patterns. *Computer*, 34:94–95, July 2001.

[18] D. Oberle, B. Berendt, A. Hotho, and J. Gonzalez. Conceptual user tracking. In E. M. Ruiz, J. Segovia, and P. S. Szczepaniak, editors, *AWIC*, volume 2663 of *Lecture Notes in Computer Science*, pages 155–164. Springer, 2003.

[19] Y. H. Park and P. S. Fader. Modeling browsing behavior at multiple websites. *Marketing Science*, pages 280–303, 2004.

[20] R. Stühmer, D. Anicic, S. Sen, J. Ma, K.-U. Schmidt, and N. Stojanovic. Lifting events in rdf from interactions with annotated web pages. In *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, pages 893–908. Springer-Verlag, 2009.

[21] G. Stumme, A. Hotho, and B. Berendt. Usage mining for and on the semantic web. In *Next Generation Data Mining. Proc. NSF Workshop*, pages 77–86, Baltimore, 2002.

[22] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the 21. International Conference on Machine learning*, NY, USA, 2004. ACM.

[23] M. Tvarožek, M. Barla, and M. Bieliková. Personalized presentation in web-based information systems. In *Proceedings of the 33rd conference on Current Trends in Theory and Practice of Computer Science*, pages 796–807, Berlin, Heidelberg, 2007. Springer-Verlag.

[24] M. Vanzin, K. Becker, and D. D. A. Ruiz. Ontology-based filtering mechanisms for web usage patterns retrieval. In *EC-Web'05*, pages 267–277, 2005.

[25] H. Yilmaz and P. Senkul. Using ontology and sequence information for extracting behavior patterns from web navigation logs. In *Data Mining Workshops*

(ICDMW), 2010 IEEE International Conference on, pages 549 –556, dec. 2010.