A Scalable Approach for Efficiently Generating Structured Dataset Topic Profiles

Besnik Fetahu[†], Stefan Dietze[†], Bernardo Pereira Nunes[*], Marco Antonio Casanova[*], Davide Taibi[‡], and Wolfgang Nejdl[†]

[†]L3S Research Center, Leibniz Universität Hannover, Germany
{fetahu, dietze, nejdl}@L3S.de
[*] Department of Informatics - PUC-Rio - Rio de Janeiro, RJ - Brazil
{bnunes, casanova}@inf.puc-rio.br
[‡] Institute for Educational Technologies, CNR, Palermo Italy
davide.taibi@itd.cnr.it

Abstract. The increasing adoption of Linked Data principles has led to an abundance of datasets on the Web. However, take-up and reuse is hindered by the lack of descriptive information about the nature of the data, such as their topic coverage, dynamics or evolution. To address this issue, we propose an approach for creating linked dataset profiles. A profile consists of structured dataset metadata describing topics and their relevance. Profiles are generated through the configuration of techniques for resource sampling from datasets, topic extraction from reference datasets and their ranking based on graphical models. To enable a good trade-off between scalability and accuracy of generated profiles, appropriate parameters are determined experimentally. Our evaluation considers topic profiles for all accessible datasets from the Linked Open Data cloud. The results show that our approach generates accurate profiles even with comparably small sample sizes (10%) and outperforms established topic modelling approaches.

Keywords: #eswc2014Fetahu, Profiling, Metadata, Vocabulary of Links, Linked Data

1 Introduction

The emergence of the Web of Data, in particular Linked Open Data (LOD) [3], has led to an abundance of data available on the Web. Data is shared as part of datasets and contains inter-dataset links [17], with most of these links concentrated on established reference graphs, such as DBpedia [1].

Linked datasets vary significantly with respect to represented resource types, currentness, coverage of topics and domains, size, used languages, coherence, accessibility [7] or general quality aspects [11]. The wide variety and heterogeneity of these dataset aspects pose significant challenges for data consumers when attempting to find useful datasets without prior knowledge of available datasets. Hence, a large proportion of datasets from the LOD cloud[1] has been overlooked in favor of well-known datasets like DBpedia or YAGO [19].

[1] `http://datahub.io/group/lodcloud`

To facilitate search and reuse of existing datasets, descriptive and reliable metadata is required. However, as witnessed in the popular dataset registry DataHub[2], dataset descriptions are often missing entirely, or are outdated, for instance describing unresponsive endpoints [7]. This issue is partially due to the lack of automated mechanisms for generating reliable and up-to-date dataset metadata, which hinders the retrieval, reuse or interlinking of datasets. The dynamics and frequent evolution of datasets further exacerbates this problem, calling for scalable and frequent update mechanisms of respective metadata.

In this work, we address the above described challenge of automatically describing linked datasets with the goal of facilitating dataset search and reuse. This paper proposes an approach for creating structured dataset profiles, where a profile describes the topic coverage of a particular dataset through a weighted graph of selected DBpedia categories. Our approach consists of a processing pipeline that combines tailored techniques for dataset sampling, topic extraction from reference datasets and topic relevance ranking. Topics are extracted through named entity recognition (NER) techniques which use reference datasets and then scored according to their relevance for a dataset based on graphical models like *PageRank* [6], *K-Step Markov*[20], and *HITS* [15]. Although this is a computationally expensive process, we experimentally identify the parameters which enable a suitable trade-off between representativeness of generated profiles and scalability. Finally, generated dataset profiles are exposed as part of a public structured dataset catalog based on the *Vocabulary of Interlinked Datasets* (VoID[3]) and the newly introduced vocabulary of links (VoL)[4]. During our experimental evaluation, dataset profiles were generated for all LOD cloud datasets which were responsive at the time of writing and our approach showed superior performance to established topic modelling techniques.

Our main contributions consist of (i) a scalable method for efficiently generating structured dataset profiles, combining and configuring suitable methods for NER, topic extraction and ranking as part of an experimentally optimised configuration, and (ii) the generation of structured dataset profiles for a majority of LOD cloud datasets according to established dataset description vocabularies.

The remainder of the paper is structured as follows. Section 3 describes the automated processing pipeline to create and expose datasets profiles. Section 4 shows the experimental setup, with the datasets and baselines used, along with the generation of the ground truth and Section 5 presents the results and their discussion. Section 6 reviews related literature. Finally, Section 7 presents the conclusion and future work.

2 Problem Definition

This section introduces and formalises the used notions of *dataset profiling*. Recall that an *RDF statement* is a triple of the form $\langle s, p, o \rangle$, where s is the *subject*

[2] `http://www.datahub.io`
[3] `http://vocab.deri.ie/void`
[4] `http://data.linkededucation.org/vol/`

(an RDF URI reference or a blank node), p is the *property*, and o is the *object* (a URI, a literal or a blank node) of the triple, respectively.

A *resource instance* r is a set of triples and is identified by a URI s. The resource *type* is determined by the triple $c = \langle s, \texttt{rdf:type}, o \rangle$. A literal l *describes* a resource instance r **iff** there exists a triple of the form $\langle s, p, l \rangle$. Given a set of datasets $\mathbf{D} = \{D_1, \ldots, D_n\}$, we denote the set of resource instances $R_i = \{r_1, \ldots, r_k\}$ and resource types $C_i = \{c_1, \cdots, c_k\}$ for $D_i \in \mathbf{D}$ $(i = 1, \ldots, n)$ by $\mathbf{R} = \{R_1, \ldots, R_n\}$ and $\mathbf{C} = \{C_1, \ldots, C_n\}$, respectively.

A *reference dataset* or *knowledge base* $\mathcal{R}$ represents a special case of a dataset D by providing a topic vocabulary. We distinguish two resource types in $\mathcal{R}$, $C = \{\texttt{entity}, \texttt{topic}\}$. An instance e of type `entity` has a literal l describing its *label* (e.g. $\langle e, \texttt{rdfs:label}, l \rangle$) and at least one triple that refers to an instance of type `topic` describing its topic. On the other hand, an instance t of type `topic` is described with a literal l, i.e. the topic label (e.g. $\langle t, \texttt{rdfs:label}, l \rangle$). In our work, DBpedia is used as reference dataset where DBpedia entities and categories represent entity and topic instances.

The set of entities $E_k = \{e_1, \ldots, e_m\}$ of a specific resource $r_k \in R_i$ of D_i (for $i = 1, \ldots, n$) is extracted through a named entity recognition function applied to literal values from r_k. The set of corresponding topics $T_k = \{t_1, \ldots, t_q\}$ for r_k is computed by accumulating all objects indicated by triples of the form $\langle e_j, \texttt{dcterms:subject}, t \rangle$ (for $j = 1, \ldots, m$). Consequently, $\mathbf{T} = \{t_1, \ldots, t_p\}$ corresponds to the set of topic classifications for all resource instances $\forall r \in \mathbf{R}$.

A *profile graph* is a labelled, weighted and directed bipartite graph $\mathcal{P} = (\sigma, \varphi, \Delta)$, where $\sigma = \mathbf{D} \cup \mathbf{T}$, and $\varphi = \{\langle D, t \rangle | D \in \mathbf{D} \wedge t \in \mathbf{T}\}$ is a set of edges between datasets and topic classifications, extracted from $\mathcal{R}$. Finally, Δ is a function that assigns an *edge weight* for each edge in φ. Correspondingly for a dataset D_k a *dataset profile graph* $\mathcal{P}_{D_k}$ represents a sub-graph of $\mathcal{P}$, hence, $\sigma = D_k \cup \mathbf{T}$, and $\varphi = \{\langle D_k, t \rangle | t \in \mathbf{T}\}$.

3 Profiling of Linked Datasets

In this section, we provide an overview of the processing steps for generating structured dataset profiles. The main steps shown in Figure 1 are the following: (i) dataset metadata extraction from DataHub; (ii) resource type and instance extraction; (iii) entity and topic extraction; (iv) topic filtering and ranking; and (v) dataset profile representation. Step (i) uses the CKAN API to extract dataset metadata for datasets part of the LOD-Cloud group in DataHub. Steps (ii) - (v) are explained in detail below.

3.1 Resource Type and Instance Extraction

From the extracted dataset metadata (i.e. SPARQL endpoint) from DataHub in step (i), step (ii) extracts *resource types* and *instances* via SPARQL queries[5] that conform to the definition of *resource types* and *instances* in Section 2.

[5] `http://data-observatory.org/lod-profiles/profiling.htm`

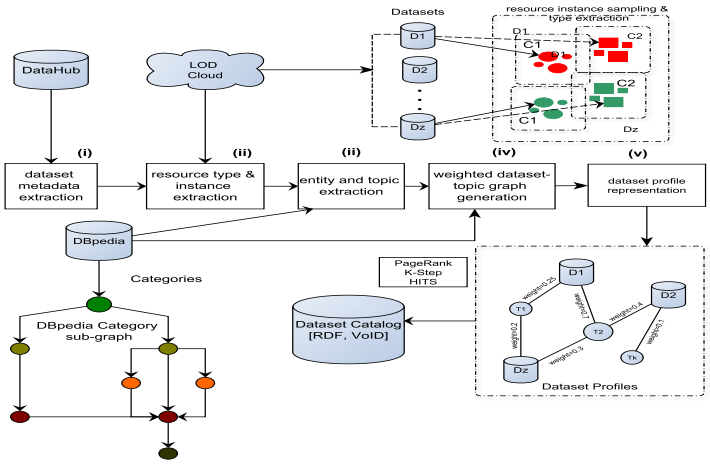


Fig. 1. Processing pipeline for generating structured profiles of Linked Data graphs.

Considering the large amount of resources per dataset, we investigate sample-based strategies as follows:

Random Sampling: randomly selects resource instances from R_i of D_i for further analysis in the profiling pipeline.

Weighted Sampling: weighs each resource as the ratio of the number of datatype properties used to define a resource over the maximum number of datatype properties over all resources for a specific dataset. The weight for r_k is computed by $w_k = |f(r_k)|/max\{|f(r_j)|\}$ $(r_j \in R_i | j = 1, \cdots, n)$, where $f(r_k)$ represents the datatype properties of resource r_k. An instance is included in a sample if, for a randomly generated number p from a uniform distribution, the weight w_k fulfils the condition $w_k > (1 - p)$. Such a strategy ensures that resources that carry more information (having more literal values) have higher chances of being included earlier at low cut-offs of analysed samples.

Resource Centrality Sampling: weighs each resource as the ratio of the number of resource types used to describe a particular resource ($V'_k \subset V_k$) divided by the total number of resource types in a dataset. The weight is defined by $c_k = |C'_k|/|C|$ with $C'_k = C \cap V'_k$. Similarly to 'weighted sampling', for a randomly generated number p, r_k is included in the sample if $c_k > (1 - p)$. The main motivation behind computing the centrality of a resource is that important concepts in a dataset tend to be more structured and linked to other concepts.

3.2 Entity and Topic Extraction

Here we describe the process of entity and topic extraction from sampled resource instances in step (ii). Recall that we use DBpedia as our reference dataset due to its broad topics coverage. To extract entities, first we combine all textual literal values of a resource (in order to provide contextual information) and consequently extract named entities from the resulting textual content using the

NER tool of choice, DBpedia Spotlight [16]. The topics $\mathbf{T}$ of sampled resources $\mathbf{R}$ represent DBpedia category instances assigned to extracted entities through the datatype property `dcterms:subject`. The topics in $\mathbf{T}$ are expanded with related topic instances (associated through datatype property `skos:broader`) up to two levels (l=2) (determined experimentally as the best expansion level, see Figure 3b).

3.3 Constructing a Profile Graph

An important step in generating the *profile graph* $\mathcal{P}$ is the ranking of associated topics in $\mathbf{T}$ for datasets in $\mathbf{D}$. Recall that $\mathcal{P}$ represents a bipartite graph, hence, a topic $t \in \mathbf{T}$ can have one or more edges connecting to datasets in $\mathbf{D}$. For instance, given two edges $\langle D_i, t\rangle$ and $\langle D_j, t\rangle$, where $D_i \neq D_j$ the computed weights $\Delta\langle D_i, t\rangle = w_i$ and $\Delta\langle D_j, t\rangle = w_j$ can be different depending on how well they represent, D_i and D_j, for $i, j = 1, \ldots, z$, respectively.

Furthermore, the function Δ relies on probabilistic graphical models. Such models are suitable as they measure the importance of each vertex with respect to other vertices in the corresponding profiles. Given a profile graph $\mathcal{P}$ and for datasets D_i, respectively its analysed resource instances are assumed to be prior knowledge. The computation of vertex weights with D_i as prior knowledge results in the computation of importance of the vertices which are part of the sub-graph connected to D_i. Consequently, this translates into computing the importance of topics $t_k \in \mathbf{T}$ $(k = 1, \ldots, n)$ with regards to D_i. Additionally, to ensure certainty of importance for D_i, the prior probability is distributed uniformly to all analysed resources in R_i, while for resources R_j from D_j the prior probabilities are set to zero.

Finally, the assigned weight to vertex t_k, with D_i as prior knowledge, infers exactly $\Delta\langle D_i, t_k\rangle$. Hence, the relationships (edges) between topic t_k and individual datasets (given as prior knowledge) have different weights, depending on the set of resources that link t_k with D_i. One of the advantages of computing the edge weights Δ is that any new dataset, which is not part of the profiles $\mathcal{P}$, can be added incrementally to the existing ones by simply computing the edge weights with its associated topics.

To illustrate why this works, consider the following example with a profile graph $\mathcal{P}$ consisting of datasets $\mathbf{D} = \{D_1, D_2\}$ with sets of resources $R_1 = \{r_{11}, r_{12}, r_{13}, r_{14}\}$ and $R_2 = \{r_{21}, r_{22}, r_{23}, r_{24}\}$, and the set of topics $\mathbf{T} = \{t_1, t_2, t_3\}$. The individual topics are associated with the following resources: $t_1 = \{r_{11}, r_{22}\}, t_2 = \{r_{11}, r_{23}, r_{24}\}, t_3 = \{r_{11}, r_{12}, r_{13}, r_{14}, r_{24}\}$. Assume we want to compute the *edge weights* between dataset D_1 and topics in $\mathbf{T}$. First, we consider D_1 as prior knowledge. Hence, we uniformly distribute the prior probability $(1/|R_1|)$ to its resources. For resources in R_2, the prior probabilities are set to zero. Finally, depending on the connectivity in the corresponding dataset profile, the topics would be ranked as follows: $\langle t_3, t_1, t_2\rangle$. The computed weights would represent the edge weights by the tuples: $\Delta\langle D_1, t_3\rangle \geq \Delta\langle D_1, t_1\rangle \geq \Delta\langle D_1, t_2\rangle$. Similarly, the *edge weights* are computed for dataset D_2.

3.4 Topic Ranking Approaches

Due to the large number of topics associated with the profile graph $\mathcal{P}$, ranking topics with respect to their relevance to datasets in $\mathbf{D}$ is crucial. A ranked set of topics enhances the usefulness of the generated profiles and facilitates the dataset recommendation and querying with higher accuracy.

Since topic extraction from the extracted entities is prone to noise from non-accurately disambiguated entities, we compute a *Normalised Topic Relevance (NTR)* score. *NTR* is a variant of the well-known *tf-idf* measure and is used to filter out noisy topics. In combination with other topic ranking approaches, it is used to determine the ideal topic expansion level. The topic rankings (edge weights) are computed through the *PageRank, K-Step Markov* and *HITS* [6, 15] graphical models, applied to the *profile graph*. The adoption of the graphical models is discussed in what follows.

Normalised Topic Relevance (NTR): The *NTR* score is an important step for pre-filtering noisy topics as a result of non-accurate entity extraction. It is computed by taking into account (i) the number of entities $\Phi(t, D)$ assigned for a topic t within a dataset D and that of entities $\Phi(t, \cdot)$ across all datasets $\mathbf{D}$ and (ii) the number of entities $\Phi(\cdot, D)$ assigned to a dataset D and for datasets in $\mathbf{D}$ $\Phi(\cdot, \cdot)$. Topics are filtered out if they have a score below a given threshold:

$$NTR(t, D) = \frac{\Phi(\cdot, D)}{\Phi(t, D)} + \frac{\Phi(\cdot, \cdot)}{\Phi(t, \cdot)}, \ \ \forall t \in \mathbf{T}, \ D \in \mathbf{D} \tag{1}$$

PageRank with Priors: is a variant of the PageRank [6] algorithm (Equation 2) that, given a data graph, in this case a dataset profile $\mathcal{P}_{D_k}$ for dataset $D_k \in \mathbf{D}$, computes the importance of dataset-topic edge weights, for each $t \in \mathbf{T}$ such that there is an edge $\langle D_k, t \rangle$. The computation of edge weights $\Delta\langle D_k, t \rangle$ is biased towards the resource instances $r \in R_k$ of D_k. Hence, the importance of a topic t is highly influenced by its connectivity with resource instances in R_k. Prior knowledge is the analysed resource instance $r \in R_k$ with prior probabilities assigned as the ratio $1/|R_k|$, while for the remaining vertices a probability of zero is assigned.

$$\pi(t)^{(i+1)} = (1 - \beta) \left(\sum_{u=1}^{d_{in}(t)} p(t|u)\pi^{(i)}(u) \right) + \beta p_t \tag{2}$$

where, t is a topic such that $\langle D_k, t \rangle \neq \emptyset$, part of the dataset profile $\mathcal{P}_{D_k}$. β is the probability of jumping back to vertices that are a priori known, $r \in R_k$. $\pi(t)$ quantifies the relative importance of t w.r.t vertices in $\mathcal{P}_{D_k}$ and is biased towards the prior knowledge $r \in R_k$. The summation in the equation quantifies the importance of t relative to vertices that have incoming connections (resource instances classified with t), $d_{in}(t)$.

HITS with Priors: although similar to PageRank with Priors, it represents a slightly different approach. The flow of visiting one vertex depends on a randomly generated binary value, where in cases it is zero it visits a vertex from an in-link for an even step, while for an odd step it follows an out-link. Otherwise, it visits one of the given vertices in R_k. As we cannot distinguish between *hubs* and *authoritative* vertices from the set of topics $t \in \mathbf{T}$ (due to their equivalent

importance), the process is simplified by having no *hub* or *authoritative* vertices.

$$a^{(i+1)}(t) = (1-\beta)\left(\sum_{u=1}^{d_{in}(t)} \frac{h^{(t)(u)}}{\sum_{t\in\mathbf{T}}\sum_{u=1}^{d_{in}(t)} h^{(i)}(t)}\right) + \beta p_t \tag{3}$$

$$h^{(i+1)}(t) = (1-\beta)\left(\sum_{u=1}^{d_{out}(t)} \frac{a^{(t)(u)}}{\sum_{t\in\mathbf{T}}\sum_{u=1}^{d_{out}(t)} a^{(i)}(u)}\right) + \beta p_t \tag{4}$$

K-Step Markov: the previous approaches represent *Markov Chains* in which the number of steps taken from the *random walk* is stochastic. *K-Step Markov* limits the number of steps to `K`. That is, the random walk starts for the given vertices of interest $t \in \mathbf{T}$ and stops after `K` steps. For a large enough `K`, the result of the ranking converges to the limit of *PageRank*. The main advantage of such an approach is scalability for large data graphs. On the other hand for step sizes not large enough the ranking lacks accuracy.

3.5 Dataset Profile Representation

The resulting profiles $\mathcal{P}$ are represented in RDF using the VoID vocabulary and are publicly available according to Linked Data principles[6]. However, VoID alone does not provide the representativeness required to capture the computed topic ranking scores. Hence, the complementary Vocabulary of Links (VoL) is introduced to complement the dataset description with a set of links to the associated dataset topics, the used ranking method and the respective score. Thus, we enable queries to select relevant datasets for a given topic. For further details, we refer the reader to the website at `http://data-observatory.org/lod-profiles/`

4 Experimental Setup

This section describes the experimental setup used for the evaluation of our approach and data. We introduce the used data, the *ground truth* and *evaluation metrics* used to measure the profiling accuracy, and the *baseline* approaches for comparison. Furthermore, the use of DBpedia in our experimental setup, for extracting structured information for the dataset profiles, does not present a limitation on using other more specialised reference dataset or a combination of reference datasets.

4.1 Data and Ground Truth

In our experiments we covered all LOD Cloud datasets whose endpoints were available. This resulted in 129 datasets with approximately 260 million resource instances and billions of triples.

[6] `http://data-observatory.org/lod-profiles/sparql`

For the evaluation we considered a subset of datasets for which we have constructed a *ground truth*[7] in the form of dataset profiles. For this task, we have exploited crowd-sourced, topic profiles already available from existing datasets. Several datasets provide a sufficient amount of manually assigned topic indicators for their resources (not the datasets themselves). These are represented by *keywords* (in the case of bibliographic resource metadata) or *tags* (for user-generated content metadata). We exploit these topic indicators, usually assigned by domain experts, to generate dataset profiles. To link such term-based topic indicators to DBpedia categories, we manually extracted entities (and eventually categories) for each topic indicator, unless the topic indicators were already available in the form of DBpedia entities. Queries to DBpedia were used to retrieve candidate entities where matching ones were selected manually. The resulting topics were ranked according to their accumulated frequency from all resources within a dataset. This is assumed to provide a more representative dataset profile.

Table 1 shows for each dataset the number of resources and the datatype properties from which topic indicators were extracted. However, due to non-accurately extracted entities, we manually checked for correctness of the named entity recognition process.

Dataset-ID	Properties	#Resources
yovisto	`skos:subject`, `dbpedia:{subject, class, discipline, kategorie, tagline}`[8]	62879
oxpoints	`dcterms:subject,dc:subject`	37258
socialsemweb-thesaurus	`skos:subject`, `tag:associatedTag`, `dcterms:subject`[9]	2243
semantic-web-dog-food	`dcterms:subject, dc:subject`	20145
lak-dataset	`dcterms:subject, dc:subject`	1691

*The datasets are accessible under: `http://datahub.io/dataset/DATASET_ID`

Table 1. Entity and Category from annotated resource instances with topic indicators for the specific datatypes properties (in the form of *keywords, tags, subjects*) for the ground truth datasets.

4.2 Evaluation Metrics

The profiling accuracy of the generated dataset profiles is measured using the `NDCG` metric (normalised discounted cumulative gain). It takes into account the ranking of topics generated using the methods in Section 3.4 compared to the ideal ranking indicated by the *ground truth*. The computation of `NDCG` is shown in Equation 5.

$$NDCG@l = \frac{DCG@l}{iDCG@l} \quad \text{where} \quad DCG@l = rel_1 + \sum_{i=2}^{l} \frac{rel_i}{log_2 i} \tag{5}$$

[7] `http://data-observatory.org/lod-profiles/ground-truth`
[8] `http://dbpedia.org/property/`
[9] `http://www.holygoat.co.uk/owl/redwood/0.1/tags/`

where $DCG@l$ represents the discounted cumulative gain at rank l, whereas $iDCG@l$ is the ideal $DCG@l$ computed from the *ground truth*.

Note that, the set of topics from the computed dataset profiles and the ones from the *ground truth* are overlapping, but not identical. Hence, for the cases where topics from the dataset profiles do not exist in the ranked set of topics in our *ground truth*, we set the ranking value to zero.

4.3 Baselines

As baselines, we chose well established approaches for topic detection. The baselines of choice generate a *profile graph* based on (i) simple *tf-idf* term weighting and (ii) *LDA* topic modelling[10] tool. In order to generate profiles consisting of DBpedia categories according to our definition from the sets of terms generated by the baselines, we followed the same approach as in Section 3.2. For the baselines, we consider the full set of resource instances for analysis. The output of each method is a set of ranked terms:

tf-idf: as the standard term frequency weighting in Information Retrieval. For *tf-idf* we assessed several initialisations of top ranked included terms (excluding stop words) $\{50, 100, 150, 200\}$, sorted based on their score. Relating to standard usage of *tf-idf*, each resource instance represents a document.

LDA: produces terms describing topics using machine learning approaches. As in the case of *tf-idf*, we use several initialisations with varying number of topics and terms defining a topic. The number of topics are $\{10, 20, 30, 40, 50\}$, with various numbers of terms per topic $\{50, 100, 150, 200\}$. The datasets are represented as single documents, since the decomposition into resource instances as documents does not influence the topic modelling tool.

The generated and ranked terms from the corresponding baseline approaches are used as seeds to generate dataset profiles. For each individual term a DBpedia entity is extracted, when there is a match from the automatic NER process. From the extracted entities, we construct the dataset profiles by taking the corresponding DBpedia categories assigned to the property `dcterms:subject` and additionally expand with equivalent broader categories. Finally, the edge weights in the profile graph $\mathcal{P}$ consist of topic scores assigned for the individual datasets and correspond to the *term weight* (computed from one of the baselines).

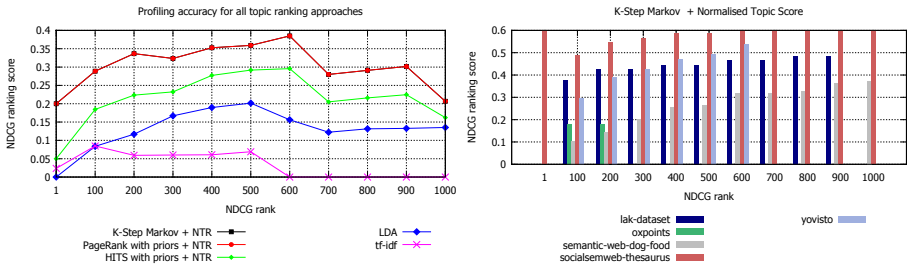
5 Results and Evaluation

In our experimental evaluation, we focus on two aspects: i) *profiling accuracy* which assesses the topic rankings induced by the graphical-models and baselines against those in the *ground truth*, and ii) *scalability* of the profiling approach finding the right trade-off between profiling accuracy and computation time.

5.1 Profiling Accuracy

In this section, we compare the *profile accuracy* from our profiling pipeline in different configurations with those of the baseline approaches.

[10] `http://mallet.cs.umass.edu`



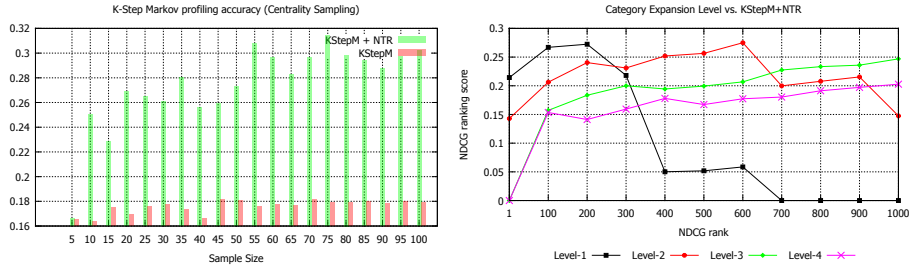
(a) (b)

Fig. 2. (a) Profiling accuracy for the different ranking approaches (in combination with `NTR`) using the full sample of analysed resource instances with `NDCG` score averaged $\forall D \in \mathbf{D}$; (b) Best performing topic ranking approach *KStepM* (in combination with `NTR`) for the full set of analysed resource instances.

The *profiling accuracy* results shown in Figure 2a are generated based on our profiling pipeline (using *PRankP*, *HITSP*, *KStepM* for topic ranking) and *tf-idf, LDA*. The results from *PRankP* and *HITSP* are generated with only 10 iterations and parameter $\beta = 0.5$, which indicates the probability of jumping back to a known vertex (in our case, an analysed resource instance of a specific dataset). For *KStepM* the number of steps was set to $K = 5$. In the case of baseline approaches we ran the experiments with several initialisations; however here we report the best performing. For *tf-idf*, the dataset profiles were generated using the top-200 terms. For the second baseline, *LDA*, we used the topic modelling tool, Mallet, with 20 topics and top-100 ranked terms. The results shown in Figure 2a correspond to an analysis conducted on the full set of resource instances. Hence, the various sampling strategies in the profiling pipeline are equal. Furthermore, the `NDCG` scores are averaged for all datasets. In the case of *PRankP, HITSP, KStepM*, the values reflect the ranking gained in combination with *NTR* as a pre-filtering step. Similarly, the results in Figure 2b show the *profiling accuracy* for the individual datasets and the best performing ranking approach, *KStepM*, where *PRankP* has comparably similar ranking with a negligible difference.

Highlighting the benefits of applying the ranking approaches in combination with *NTR*, Figure 3a shows the difference in profiling accuracy for *KStepM* approach at `NDCG@100` (averaged over all datasets) for different sample sizes. The topic scores computed by *NTR* are used to filter noisy topics, when their values are below the average from all topics in a dataset profile $\mathcal{P}_D$.

To determine the correct topic expansion level, we measure the correlation between the expansion level and *profiling accuracy* Figure 3b. The results show the impact of the expansion level on the *profiling accuracy* for the case of the topic ranking approach, *KStepM*. The intuition is that, at a certain expansion level, the dataset profiles are associated with noisy topics (when a topic is assigned to too many entities). Figure 3b shows that the highest overall ranking accuracy was achieved at the expansion level of two.



(a) (b)

Fig. 3. (a) Comparison of profiling accuracy for *KStepM+NTR* and *KStepM* at `NDCG@100`; (b) Category level expansion impact on profiling accuracy for *KStepM+NTR*.

5.2 Scalability vs. Accuracy Trade-Off: Impact of Sample Size

We analyse the impact of the various sampling strategies (*random*, *weighted* and *centrality*) at different sample sizes on ranking accuracy, to identify a suitable balance between *ranking time* and *profiling accuracy*.

To find the ideal trade-off between *scalability* and *accuracy*, we analyse the behaviour of the ranking metric `NDCG` as follows: (i) average performance (Δ`NDCG`) over all datasets and computed ranks ($l = 1, \ldots, 1000$), (ii) *profiling accuracy* and *topic ranking time*, using *KStepM* ranking approach.

For (i), Figure 4 shows the results of the Δ`NDCG` score for *KStepM* at different sample sizes (x-axis). The plot for the individual datasets shows the *standard deviation* from the average value of Δ`NDCG`, indicating the stability of the profiling accuracy. While, for (ii), Figure 5 shows the correlation between profiling accuracy and ranking time. It assesses attributes such as the amount of time taken to rank topics (*KStepM, HITSP, PRankP*) and the different sample sizes. In detail, the leftmost *y-axis* shows the log-scale of the amount of time (in seconds) it takes to rank the topics at the different sample sizes. The rightmost *y-axis* shows the profiling accuracy achieved at a specific sample size.

5.3 Discussion and Analysis

The results shown in the previous two sections support the proposed choice of steps in the profiling pipeline and identified suitable parameters. The combination of topic ranking approaches (*PRankP, HITSP* and *KStepM*) with *NTR* significantly improves the profiling accuracy. In Figure 3a and for *KStepM*, a drastic increase in accuracy can be noted for all sample sizes. This is rather expected as the *NTR* scores serve as a pre-filtering mechanism for noisy topics. From the overall ranking comparisons in Figure 2a, *KStepM* achieves the best results (Figure 2b), with *PRankP* having comparably similar results.

By contrast, the baseline ranking approaches show that the overall performance is relatively low. The *LDA*-based baseline approach achieves comparable accuracy only at rank $l = 500$. The results for the second baseline based on

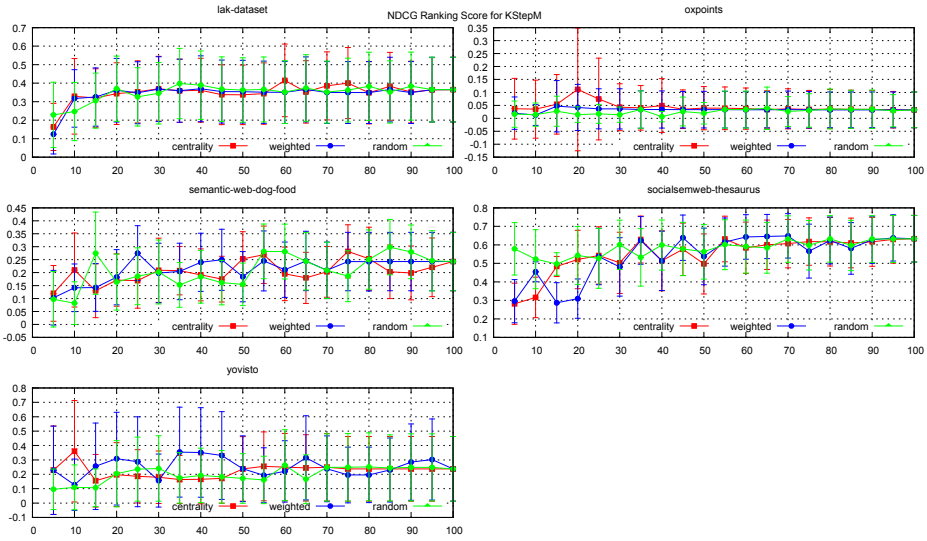


Fig. 4. Profiling accuracy averaged for all ranks $l = \{1, \ldots, 1000\}$. The graph shows the standard deviation of ΔNDCG from the expected ranking at the different sample sizes (x-axis).

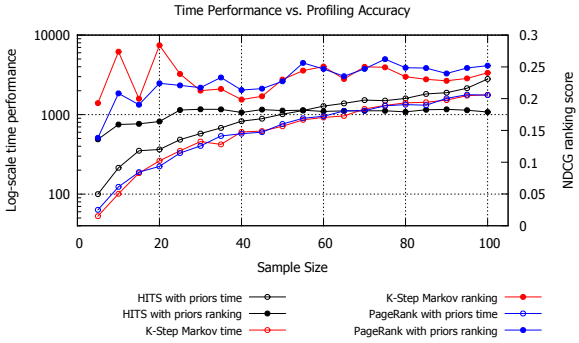


Fig. 5. The trade-off between profiling accuracy (ΔNDCG averaged over all datasets and ranks) and the topic ranking time based on the different graphical-models.

tf-idf are uniformly very low at all ranks, with most values being well below 0.1. The results from the baselines are attained from the best performing initialisations (see Section 4.3). In the case of *LDA* we used 20 topics with top-100 terms per topic, and for *tf-idf* an increase of more than top-200 analysed terms did not benefit significantly the profiling accuracy. The difference between the best performing baseline based on *LDA* and that based on *KStepM+NTR* is ΔNDCG@100=+0.21 in favour of the latter.

The results in Figure 4 show that, at low sample sizes, the accuracy is already fairly stable. In other words, the average profiling accuracy for the different

ranking approaches and sampling strategies increases slightly with the increase of sample size, while its standard deviation decreases. We could identify sample sizes of 5% and 10% as nearly optimal, which are also nearly optimal with regards to the balance between accuracy and scalability in Figure 5. The dataset profiling time is reduced significantly while aiming for a suitable trade-off between scalability and profile accuracy. The process of generating profiles contains three computationally intensive steps: (i) indexing resources for further analysis; (ii) performing the NER process; and (iii) topic ranking. With respect to (i), indexing 10% of resource instances takes on average, 7 minutes per dataset, in contrast to up to 3 hours on average when considering all resource instances. For (ii), since we use the online service of DBpedia Spotlight, the process is non-deterministic, as it is dependent on the network load and the number of simultaneous requests. Such process could be optimised by hosting the service locally. Finally, for (iii), the topic ranking process is optimised down to 2 minutes, for 10% resources, from 45 minutes, when considering the full set of resources (Figure 5).

Finally, the fluctuations in profiling accuracy in Figure 4 show high deviations for dataset '`oxpoints`'. This can be explained by the fact that its resources contain geo-information about the University of Oxford and as such it presents a difficult case due to the low coverage from DBpedia content.

6 Related Work

Although no approach considers specifically the problem of generating metadata about Linked Data sets profiles, our approach is closely related to a LOD Cloud dynamics of changes analysis [13]. The work in the reported paper is related to several fields ranging from VoID data generation [5, 4], semantic indexing [18], graph importance measures [20, 12], and topic relevance assessment [8, 9] address similar problems. Thus, in this section, we briefly review the literature and compare our approach with related literature.

Generating VoID data about Linked Data sets is considered in [5], where individual triples are analysed and, based on commonly shared predicates, the corresponding datasets are clustered. In a later work, Böhm et al. [4] cluster resources based on the dataset specific ontologies used by considering the relationship between the different resource classes. In spite of using specific ontologies to create clusters, we use established reference datasets.

Recently, Hulpus et al. [12] proposed the *Canopy* framework that, for a given set of extracted topics from analysed textual resources, the matching DBpedia sub-graph is retrieved and the corresponding relationships are quantified using graph importance measures. In our case, we automatically extract entities from textual resources and further expand to the related DBpedia category sub-graphs. A different approach is presented by White et al. [20], where they measure the relative importance of a node in a data graph by incorporating knowledge about prior probability of a specific node. We follow the same strategy to measure the importance of topics in the generated dataset profiles.

Tìpalo, a framework introduced by Gangemi et al. [10], analyses heuristics for typing DBpedia entities using information extracted from Wikipedia pages

mentioning a specific entity. In our work, we focus on topic assessment using DBpedia graph and the context of analysed resources of a dataset from which an entity is extracted.

Another framework is *Sindice* [18], that indexes RDF documents and uses DBpedia entities as a source to actively index resources. Additionally, Kiryakov et al. [14] index the Web of Documents and capture extracted named entities in a manually crafted ontology. Comparing to our work, we go beyond mere annotations and generate an interlinked data graph of datasets based on topics which are quantified for their importance based on the support given from the individual resource instances.

Käfer et al. [13] have crawled and analysed the LOD cloud focusing mostly on the dynamics of changes in datasets (predicates, number of instances, etc). The crawling process relies on pre-selection of prominent resources (ranked based on PageRank). We aim at generating dataset profiles and analysing the temporal aspects of topics on how they evolve during time. LODStats [2] analyses the LOD-cloud structure and provides statistical characteristics of datasets and metrics related to vocabulary usage. In spite of the insights gained through such an analysis, we focus at a content-wise analysis.

7 Conclusions

In this paper, we proposed an approach to automatically generate structured dataset profiles with the overall goal of facilitating the assessment, search and discovery of LD datasets. Aiming for a method which is scalable and efficient and yet, at the same time, provides a high level of accuracy and representativeness of the generated data, our approach uses sampling techniques together with ranking methods to provide the *profile graph*. Based on experimental evaluation, the most suitable trade-off is found between small sample sizes to cater for efficiency and representativeness of the resulting profiles.

As part of our experiments, we generated dataset profiles for all datasets in the LOD Cloud. The evaluation shows that, even with comparably small sample sizes (10%), representative profiles and rankings can be generated (i.e. Δ`NDCG`=0.31 for '`socialsemweb-thesaurus`'), when applying *KStepM* combined with the *NTR*. The results demonstrate superior performance when compared to *LDA* with Δ`NDCG`=0.10 (with the full set of resource instance).

It has been noted that meaningfulness and comparability of topic profiles can be increased when considering topics associated with certain resource types only. As part of our current work we are developing resource type-specific dataset profiles and the tracking of topic profile evolution. These take advantage of our *profile graph* to provide more specific dataset search and browsing capabilities.

Acknowledgements: This work was partly funded by the LinkedUp (GA No:317620) and DURAARK (GA No:600908) projects under the FP7 programme of the European Commission.

References

1. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. G. Ives. Dbpedia: A nucleus for a web of open data. In *6th International Semantic Web Conference (ISWC)*, pages 722–735, 2007.
2. S. Auer, J. Demter, M. Martin, and J. Lehmann. Lodstats - an extensible framework for high-performance dataset analytics. In *Knowledge Engineering and Knowledge Management - 18th International Conference, Galway City, Ireland, October 8-12, 2012*, pages 353–362, 2012.
3. C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.
4. C. Böhm, G. Kasneci, and F. Naumann. Latent topics in graph-structured data. In *21st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2663–2666, 2012.
5. C. Böhm, J. Lorey, and F. Naumann. Creating void descriptions for web-scale data. *J. Web Sem.*, 9(3):339–345, 2011.
6. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30(1-7):107–117, 1998.
7. C. Buil-Aranda, A. Hogan, J. Umbrich, and P.-Y. Vandenbusshe. Sparql web-querying infrastructure: Ready for action? In *Proceedings of the 12th International Semantic Web Conference, Sydney, Australia*, 2013.
8. M. d'Aquin, A. Adamou, and S. Dietze. Assessing the educational linked data landscape. In *Web Science (WebSci)*, pages 43–46, 2013.
9. B. Fetahu, S. Dietze, B. Pereira Nunes, and M. Antonio Casanova. Generating structured profiles of linked data graphs. In *Proceedings of the 12th International Semantic Web Conference (ISWC)*. Springer, 2013.
10. A. Gangemi, A. G. Nuzzolese, V. Presutti, F. Draicchio, A. Musetti, and P. Ciancarini. Automatic typing of dbpedia entities. In *International Semantic Web Conference (ISWC)*, pages 65–81, 2012.
11. C. Guéret, P. T. Groth, C. Stadler, and J. Lehmann. Assessing linked data mappings using network measures. In *9th Extended Semantic Web Conference (ESWC)*, pages 87–102, 2012.
12. I. Hulpus, C. Hayes, M. Karnstedt, and D. Greene. Unsupervised graph-based topic labelling using dbpedia. In *ACM International Conference on Web Search and Data Mining (WSDM)*, pages 465–474, 2013.
13. T. Käfer, A. Abdelrahman, J. Umbrich, P. O'Byrne, and A. Hogan. Observing linked data dynamics. In *10th Extended Semantic Web Conference (ESWC)*, pages 213–227, 2013.
14. A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff. Semantic annotation, indexing, and retrieval. *J. Web Sem.*, 2(1):49–79, 2004.
15. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
16. P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *7th International Conference on Semantic Systems (ISWC)*, pages 1–8, 2011.
17. B. P. Nunes, S. Dietze, M. A. Casanova, R. Kawase, B. Fetahu, and W. Nejdl. Combining a co-occurrence-based and a semantic measure for entity linking. In *10th Extended Semantic Web Conference (ESWC)*, pages 548–562, 2013.
18. E. Oren, R. Delbru, M. Catasta, R. Cyganiak, H. Stenzhorn, and G. Tummarello. Sindice.com: a document-oriented lookup index for open linked data. *IJMSO*, 3(1):37–52, 2008.

19. F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In C. L. Williamson, M. E. Zurko, P. F. Patel-Schneider, and P. J. Shenoy, editors, *WWW*, pages 697–706. ACM, 2007.
20. S. White and P. Smyth. Algorithms for estimating relative importance in networks. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 266–275, 2003.