# Automatic Domain Identification for Linked Open Data

Sarasi Lalithsena*, Prateek Jain†, Pascal Hitzler* and Amit Sheth*
*Kno.e.sis Center, Wright State University, Dayton, OH, USA
{sarasi, pascal, amit}@knoesis.org
†IBM T.J. Watson Research Center, Yorktown, NY, USA
jainpr@us.ibm.com

***Abstract*—Linked Open Data (LOD) has emerged as one of the largest collections of interlinked structured datasets on the Web. Although the adoption of such datasets for applications is increasing, identifying relevant datasets for a specific task or topic is still challenging. As an initial step to make such identification easier, we provide an approach to automatically identify the topic domains of given datasets. Our method utilizes existing knowledge sources, more specifically Freebase, and we present an evaluation which validates the topic domains we can identify with our system. Furthermore, we evaluate the effectiveness of identified topic domains for the purpose of finding relevant datasets, thus showing that our approach improves reusability of LOD datasets.**

***Index Terms*—Linked Open Data Cloud, Domain Identification, Dataset search**

## I. Introduction

Linked Open Data (LOD) has gained significant visibility and adoption since its inception. Starting with 12 datasets in 2007, currently it consists of more than 300 datasets. The rapid growth in the number of LOD datasets reveals the interests of data publishers to publish their data as structured data on the data cloud and this trend is likely to continue. Furthermore, diverse range of domains and topics covered by these datasets are also increasing. Researchers and practitioners have utilized the datasets for various tasks such as Type Coercion in Question Answering [1] and Music Discovery[1].

Despite the adoption, increase in the size and diversity of the datasets creates challenges in identifying the relevant datasets for the task at hand. Even though popular datasets such as DBPedia,[2] Freebase[3] and MusicBrainz[4] are well known and widely used in the community, there are other hidden gems like Climbdata[5] and Lingvoj[6]. Climbdata provides information about climbing routes, whereas Lingvoj provides various ways different languages relate to things such as country and organization. Indeed these datasets may also be useful for certain kinds of specialized applications, however without a registry of topics, it is difficult for a potential user to find them. Therefore, as an early step towards searching and identifying relevant datasets, identifying the topic domains of a dataset is extremely important for the consumers of LOD. Here topic domains can be considered as the keywords to describe topics covered by a dataset.

Currently the main entry point for discovering and identifying new datasets is the LOD Diagram[7]. This diagram is generated based on the datasets being added to the lod-cloud group in the CKAN data hub[8]. CKAN allows data publishers to manually assign predefined sets of tags such as *media, geography, life sciences, publications, government, e-commerce, social web, user generated content, schemata* and *cross-domain* to classify the datasets to different domains. CKAN administrators manually review these assignments and have used these tags to better organize the LOD diagram. This process has a number of shortcomings, such as the following.

- The increasing diversity of the datasets makes it difficult to work with a fixed number of pre-defined tags. For an example, it is hard to decide on proper tags for the Lingvoj dataset using the predefined CKAN tags.
- With the rapidly increasing number of available datasets, the manual reviewing process will soon be unsustainable.
- Human classification is subjective and may not capture the essence and breadth of the dataset.

These shortcomings emphasize the need for more systematic and sophisticated approaches to identify the topic domains of the datasets. For this, we take a straightforward perspective on the topic identifiers needed: We use tags, which will usually be general terms such as "music," "geography," or "artist" as identifiers to describe topic domains.

Automatic topic domain identification for LOD datasets is an interesting and challenging issue due to several reasons. First, schema information plays a critical role in identifying the topic domains of a dataset, but most of the LOD datasets only contain very shallow schemas. Thus, schema information by itself will not be enough to identify topic domains. Second, people represent data that belongs to various domains in different granularities. For example, DBpedia contains information about diverse domains including music, while MusicBrainz focuses on the music domain. DBpedia and MusicBrainz use different schemas to represent data, so by looking at the

[1]http://www.bbc.co.uk/music
[2]http://dbpedia.org/About
[3]http://freebase.com/
[4]http://musicbrainz.org/
[5]http://datahub.io/dataset/data-incubator-climb
[6]http://www.lingvoj.org/
[7]http://http://lod-cloud.net/ – the latest picture is almost two years old and therefore outdated, but still it is a major entry point.
[8]http://thedatahub.org/group/lodcloud

schemas it is a nontrivial task to identify music as a common domain covered by both datasets.

In this work, we provide an approach to automatically identify the topic domains of datasets by utilizing knowledge sources in other LOD datasets, such as the hierarchy within Freebase. We believe community driven knowledge sources and their hierarchy will enable us to cover a wide variety of domains, and in fact this type of "bootstrapping" of LOD has been used before for other purposes [2]. Also, we present a search application built on the identified domains to search and identify relevant datasets within LOD. Furthermore, we provide an evaluation to validate the domains identified and also evaluate the effectiveness of the identified domains for searching datasets in comparison to existing systems.

The rest of paper is organized as follows. Section II presents our approach for identifying topic domains for datasets and Section III presents implementation details in brief. In Section IV we present the evaluation and Section V discusses related work. We conclude in Section VI.

## II. Approach

Our approach provides a technique to automatically identify the main topics of LOD datasets by utilizing Freebase as both background knowledge and to provide the vocabulary for the topic tags. We will in particular make use of the fact that each Freebase instance or article (we will call them *Freebase instances* in the following), is assigned one or more *Freebase types* within Freebase (such as *mountain*). Each of these types, in turn, is assigned to a *Freebase domain* (such as *geography*).

In a nutshell, our approach is based on assigning Freebase types and domains to the instances in an input LOD dataset, together with a weight compute from its frequency count. While we have developed our approach with Freebase in mind, and we describe it as such below, it will be clear from the description that our approach is adaptable to other settings. We will discuss this further in the conclusions.

In more detail, our approach consists of the subsequent steps explained below in Sections II-A to II-D. Figure 1 depicts the workflow of our approach with examples at each step.

### A. Category Identification

#### 1) Instance Identification

The topic domains of a dataset are implicitly determined by the collection of entities it contains. As an example, the domain of 'GeoNames'[9] is predominantly geo-spatial because it contains a large number of geo-spatial entities such as countries, cities and villages. Therefore, our approach primarily utilizes the instances of the dataset in conjunction with type information of the instances to identify the topic domains of each dataset.

As first step, the input dataset is processed to retrieve (i) the instances, (ii) their corresponding labels or human readable values, (iii) classes of the instances, and (iv) class name labels or human readable values.

After this, the next step is the identification of corresponding or closely related Freebase instances for all instances of the input dataset. This is achieved by utilizing the labels of the instances in the dataset combined with the concept names to which the instance belongs,[10] and executing a search on Freebase using its API[11]. The combination of instance labels with concept labels can improve the accuracy of the approach since in some cases instance labels by themselves return irrelevant results. For example, consider the instance 'Ignimbrite' from the Climb Dataincubator dataset[12]. Using 'Ignimbrite' as the search string on Freebase leads to multiple hits such as 'Ignimbrite' and the book 'Geology of a Miocene ignimbrite layer'. Appending the type information, i.e., 'Rock', to the query term enhances the precision by eliminating the book 'Geology of a Miocene ignimbrite layer'. The instance name alone is utilized as the search term where type information does not retrieve any results.

This step is illustrated in Step 1.1 of the Figure 1 for three different instances.

#### 2) Category Hierarchy Creation

The search results, i.e., the identified Freebase instances for each query from the previous step, are used to obtain what we call *category hierarchies* for them: The Freebase search API is used to identify the Freebase types within which the Freebase instances have been categorized, and we also keep track of the corresponding Freebase domains. For the term *Ignimbrite*, for example, the Freebase API returns the type 'rock_type' in the domain 'geology', giving rise to the category hierarchy consisting of 'rock_type' and 'geology'.

At this point, the system has generated a set of category hierarchies for each given instance of the dataset, as shown in Step 1.2 of Figure 1.

### B. Category Hierarchy Merging

Once the category hierarchies have been created, this step merges all of those with the same Freebase domain, by creating a tree of depth 2 with the domain as root and the types as leaves. Step 2 in Figure 1 shows the resulting category hierarchies for the instances (a) and (b) after merging. The two category hierarchies with domain 'geography' from instance (a) have been merged. This step is repeated for the two hierarchies with domain 'music' of instance (b). This step results in a forest-like data structure with a number of category hierarchy trees rooted at a common generic node.

### C. Candidate Category Hierarchy Selection

At the end of the previous step the input LOD dataset now has multiple category hierarchy trees associated with it, due to the varied collection and classification of instances. However, not all of these hierarchies are relevant and/or significant for a given dataset. Therefore, this step in our approach filters out insignificant category hierarchies by using a simple heuristic. Given a concept $C$ of the input dataset, each instance of $C$ gives rise to several category hierarchy trees as results of the previous steps. We now identify Freebase domains which occur most often as roots of these trees, and retain only the trees with these roots, discarding all others.

[9] http://www.geonames.org/ontology/documentation.html

[10] i.e., to which it is explicitly assigned; we do not consider inferred types.

[11] https://developers.google.com/freebase/

[12] http://climb.dataincubator.org/

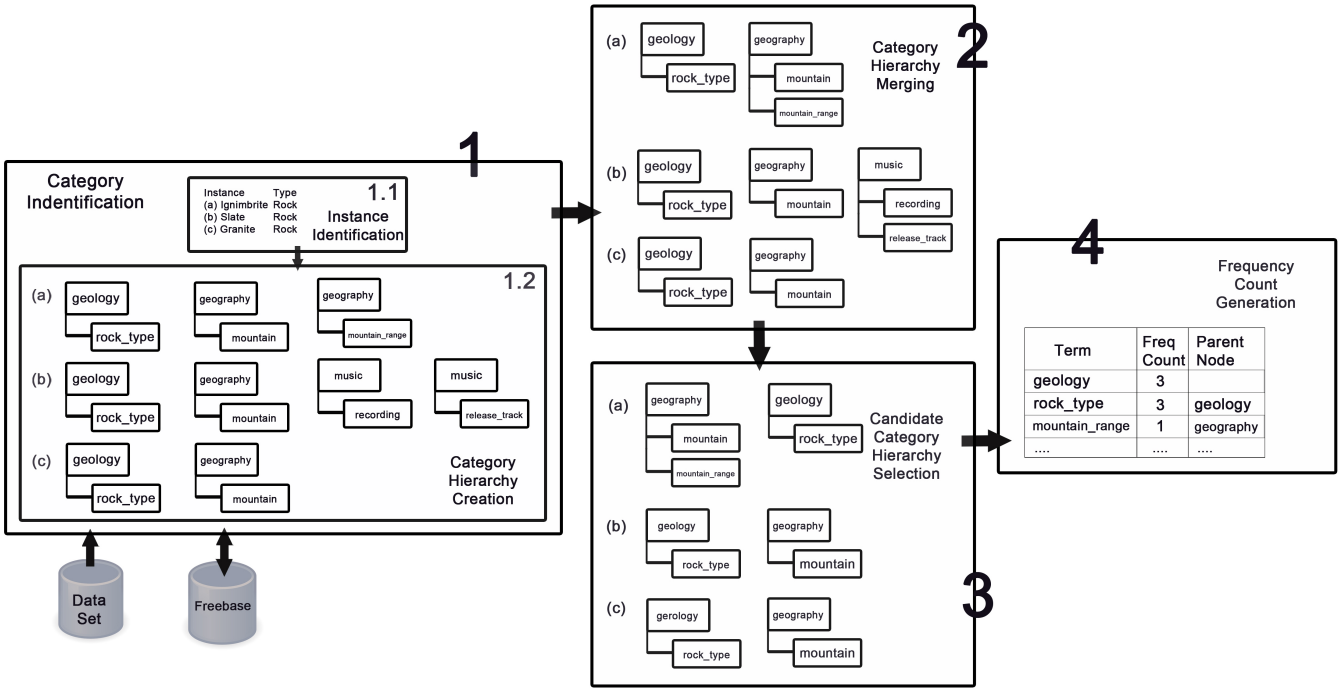


Fig. 1. *Workflow for identifying topics*

As an example, consider the 25 instances of type 'Rock' in the Climb Incubator dataset. Our system generates multiple category hierarchies for the 25 instances. 22 out of 25 instances have 'geology' as the root of the hierarchy, while 3 have 'music' as the root node. Using simple majority as deciding mechanism, all hierarchies with 'geology' as root are retained and all hierarchies with 'music' as root are discarded.

This is illustrated in Step 3 in Figure 1 for a small example consisting of only three instances. Category hierarchies rooted at 'geology' and 'geography' are retained while the one with 'music' as root is removed. This process greatly reduces the impact made by false positives returned by the search API.

### *D. Frequency Count Generation*

The next step involves assigning a frequency count to each of the terms in the resulting category hierarchies to describe their relative importance with regard to a given dataset. This count is generated by considering all category hierarchies from all instances of the dataset: Given an input LOD dataset $D$ and a Freebase type or domain $T$, let $\mathcal{H}$ be the set of all category hierarchies generated from $D$ using the steps described above, and let $\text{Freq}_D(T)$, called the *frequency count of $T$ for $D$*, be the number of occurrences of $T$ in $\mathcal{H}$.

Note that the frequency count is generated both for the root node and for all child nodes occurring in category hierarchies. The Table given in Step 4, Figure 1 shows frequency counts calculated for 'geology', 'rock_type' and 'mountain_range' for our example. These terms which consist of Freebase domains and types can be considered as the topic domains for a given dataset. A higher frequency count for a term provides an evidence for the term being a good descriptor for the dataset, because it shows that a large number of instances can be described by the given term.

## III. Implementation

Our system has been implemented in Java using Jena[13] and the Freebase API. In order to scale to large datasets, our system has been deployed on a Hadoop cluster[14] consisting of 15 nodes, using a Map-Reduce job. The list of instance and type labels collected from the dataset is given as input to the Mapper task as pairs <InstanceLabel, TypeLabel>. The Mapper task performs the category hierarchy building by querying the knowledge base, and merges the category hierarchies as described in Step 2 in Figure1. The Mapper writes its output as <TypeName, CategoryHierarchies> for each instance. Here CategoryHierarchies refer to the hierarchies generated in Step 2 in Figure1. Once the Mapper tasks are done, the Reducer initializes its task by taking TypeName as the key, i.e., all the instances with the same TypeName will be performed by a single reducer. The Reducer processes candidate category hierarchy selection, performs frequency count generation and keeps track of the root nodes associated with non-root nodes.

## IV. Evaluation

In order to evaluate our approach, we ran our system on 30 LOD datasets which cover variety of domains. These datasets include some prominent ones such as BBCMusic, DailyMed, VIVO Indiana, LinkedMovieDB and SemanticWebDogFood.

[13]http://jena.apache.org/
[14]http://hadoop.apache.org/

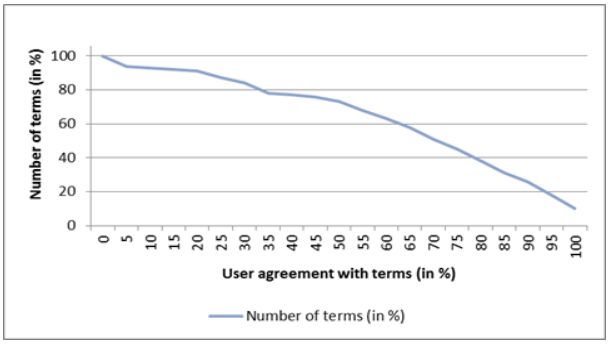


Fig. 2. *User agreement on appropriateness of terms.*

| **Dataset** | **Term** | **%** | **Dataset** | Term | **%** |
|---|---|---|---|---|---|
| BBCMusic | music* | 100 | BBCProgram | TV | 100 |
| BBCWildLife | animal* | 100 | Climbdata | location* | 85 |
| DailyMed | medicine* | 90 | DBTuneClassic | music* | 100 |
| Diseasome | disease* | 100 | DrugBank | medicine* | 100 |
| Eumida | university | 80 | EuroStat | location* | 95 |
| Foodalista | food* | 100 | GeneBank | gene | 100 |
| GeoSpecies | biology* | 95 | Lingvoj | language* | 95 |
| EnergyData | organization | 50 | LMDB | film* | 100 |
| NASA | spaceflight* | 100 | ordinanceSurvey | citytown | 95 |
| semwebdogFood | people | 65 | UKPatentInfo | organization* | 65 |
| VIVOIndiana | organization | 80 | WorldFactBook | country | 90 |
| Airport | aviation | 100 | ECSRKB | people* | 65 |
| EUInstitutions | organization* | 95 | SIDER | drug | 85 |
| FarmersMarket | location* | 75 | Medicare | Medicine* | 100 |
| Gutenburg* | book* | 65 | Telegraphic | location* | 85 |

TABLE I
TERMS WITH HIGHEST USER AGREEMENT FOR EACH DATASET. WE INDICATE BY A STAR (*) THAT A TERM WAS ALSO HIGHEST RANKED BY OUR SYSTEM.

The identified topic domains for each dataset can be found at http://knoesis.org/LODSearch/topics.html.

In order to evaluate the quality of our approach, we use two different settings. The first setting (see Section IV-A) aims to validate the domains we identified involving users as subjects, and the second setting (see Section IV-B) evaluates the identified domains in terms of their effectiveness for finding LOD datasets on given topics.

### A. Appropriateness of identified domains

Since there is no existing benchmark for this purpose we validate the identified domains using human subjects. To do so, we extracted the two highest ranked topic domains from the set of roots (Freebase domains) and the two other highest ranked terms from the leaves (Freebase type) for each dataset. Then we mixed these with four other random Freebase types/domains from the Freebase hierarchy. The reason behind selecting terms from both roots and leaves instead of taking just the four highly ranked terms is to ensure that the terms cover more than one Freebase domain. This allows us to assess the validity of assigning more than one Freebase domain as a topic domain. We then presented these eight terms to each of twenty users and asked them to select the terms that best represent the topic domains of the dataset. Of the 20 people, 10 people responded to our request for participation on W3C Semantic Web and LOD mailing lists[15] and an other 10 were members of two different organizations, DERI and Knoesis, who work with LOD datasets and are familiar with them.

To summarize the results, we calculated the percentage of users which agree for each term generated by our approach. The graph in Figure 2 shows how many users agreed on how many terms being appropriate descriptors, from a total of 20 users (=100%, horizontal axis) and 120 terms (=100%, vertical axis). The data shows that 50% of the users agreed on 73% (88 out of 120) of the terms being appropriate descriptors. Table I shows in more detail results for the terms which had the highest user agreement for each dataset.

Even though the system performs well with most datasets, with some datasets such as LinkedEnergyData and UK-PatentInfo our approach fails to identify the prominent topic domains. The more likely reason for this is the lack of matching Freebase instances for the entities in these datasets.

### B. Usefulness of identified domains for dataset search

In this section we present comparative evaluations of our approach by demonstrating its effectiveness for finding LOD datasets, compared with (1) a baseline obtained by a user study employing existing LOD lookup services such as semantic search engines (Section IV-B1), and (2) searching on the CKAN data hub repository (Sections IV-B2 and IV-B3).

For the evaluation, we created a LOD dataset search application based on the identified topic domains, which is available at http://knoesis.org/LODSearch/. The application makes use of an index, more specifically it leverages the terms and statistical information collected during our process of topic domain identification. Each term is indexed with a list of datasets ranked by the normalized frequency count $\text{NFreq}_D(T)$ of the term. The normalized frequency count is calculated as

$$\text{NFreq}_D(T) = \frac{\text{Freq}_D(T)}{\text{Total No of Instances in } D}$$

#### 1) User study

We conducted a user study to evaluate how useful the results generated by our approach are for dataset search, compared to using CKAN, LODStats [3][16] or the Sindice semantic search engine [4]. CKAN and LODStats are two systems which allow people to identify relevant datasets based on keywords. Furthermore, CKAN uses metadata provided by users. More details on these systems are given in the Section V. There are a number of semantic web search engines such as Watson [5] and Swoogle [6]. We choose Sindice mainly because (1) it allows to group the search results by datasets which is directly relevant to our approach, and (2) it is a very recent system and regularly updated.

For the evaluation, we performed the following steps.

1) We asked four users to come up with twenty terms each that reflect some topic domains of datasets present in LOD. Table II presents the list of 20 terms for each of the 4 users. From these eighty terms selected by the

[15]http://lists.w3.org/Archives/Public/public-lod/2013May/0110.html

[16]http://stats.lod2.eu/rdfdocs

| User | Terms |
|---|---|
| User1 | music, animal, drug, gene, food, conference, spacecraft, energy, language, university, tv program, film, mountain, geology, biology, spacecraft, instrument, recipe, disease, artist |
| User2 | music, animal, drug, gene, food, conference, spacecraft, energy, language, university, tv program, rock, geology , astronaut, phenotypes, composer, recipe, country, artist, organism |
| User3 | music, animal, drug, food, conference, spacecraft, energy, language, university, tv program, invention, book, geology, biology, phenotypes, composer, student, location, researcher, region |
| User4 | music, animal, drug, gene, food, conference, energy, language, university, patent, film, book, geography, biology, instrument, student, astronaut, disease, artist, nasa |

TABLE II
TERMS SELECTED BY USERS TO DESCRIBE THE DOMAIN

users, 20 terms were selected which were most often mentioned.

2) These twenty terms were used in order to evaluate our approach compared with CKAN, LODStats and Sindice. We retrieved the top ten results for each term for all systems. The results for all the terms can be found at the web page we used for the evaluation.[17]
3) The results for each term and each system were presented to 27 different users and they were asked to identify which set of results they preferred the most. The familiarity of the users with the LOD datasets varied from medium to expert. The results were provided to the users in a blind fashion, i.e., the users were not provided with the names of the systems which generated each set of results. The users were asked to rank the four result sets from 1 (best) to 4 (worst) based on their familiarity with the datasets and expectations based on the terms.
4) We calculated a user preference score using the user rankings to assess the performance of each system for each term. The score $R(S,T)$ for term $T$ in system $S$ is calculated using the weighted average[18]

$$R(S,T) = \frac{\sum_{i=1}^{4}((5-i)*(N_{iTS}))}{\text{Total Number of Users}},$$

where $N_{iTS}$ is the number of users which rated rank $i$ for the term $T$ in the system $S$. This is essentially the most common method to summarize user ratings in product ranking systems. Note that a higher score indicates stronger performance.

Table III summarizes the results: CKAN ranked best for 12 terms while our approach ranked best for 9 terms. LODStats ranked best for 1 term. While our approach generates only second best results in some cases, it needs to be noted that our system indexes only 30 datasets, while other systems index over 290 datasets. Note, also, that CKAN uses keywords, user's metadata and manual tagging, while our system creates topic domain tags automatically, and thus scales better.

[17]http://knoesis-hpco.cs.wright.edu/LODYellowPagesEvaluation/
[18]http://en.wikipedia.org/wiki/Weighted_mean

| Term | Our Approach | CKAN | LOD Stat | Sindice |
|---|---|---|---|---|
| music | 2.037 | **3.74** | 3.11 | 1.333 |
| artist | 2.815 | **3.926** | 1 | 2.259 |
| biology | **3.481** | 3.333 | 1 | 2.185 |
| animal | 2.926 | 1.63 | **3.481** | 1.926 |
| geology | 2.852 | **3.666** | 1 | 2.481 |
| drug | 2.926 | **3.148** | 2 | 2.555 |
| gene | 2.148 | **3.333** | 3.074 | 1.222 |
| university | **3.185** | 3.148 | 2.37 | 1.222 |
| food | **3.259** | 2.296 | 3 | 1.259 |
| language | 3.148 | **3.74** | 1 | 2.11 |
| spacecraft | **4** | **4** | 1 | 2 |
| conference | 2.814 | **3.555** | 1 | 2.666 |
| astronaut | **4** | **4** | 1 | 2 |
| composer | **3.815** | 3.037 | 1 | 2.11 |
| tv program | **3.666** | 2.923 | 1 | 2.370 |
| instrument | **3.852** | 2 | 2 | 3.148 |
| recipe | **3.926** | 2 | 2 | 3.074 |
| student | 2 | **3.889** | 2 | 3.111 |
| phenotypes | 2 | **3.923** | 2 | 3.037 |
| energy | 1 | **3.74** | 3.26 | 3.03 |

TABLE III
COMPARATIVE DATASET SEARCH EVALUATION RESULTS

| Term | P | R1 | F1 | R2 | F2 |
|---|---|---|---|---|---|
| music | 0.286 | 1 | 0.445 | 0.1 | 0.148 |
| artist | 0.4 | 1 | 0.571 | 0.2 | 0.267 |
| biology | 0.125 | 1 | 0.222 | 0.333 | 0.182 |
| animal | 0* | 0* | n/a* | 0* | n/a* |
| geology | 0* | 0* | n/a* | 0* | n/a* |
| drug | 0.6 | 0.667 | 0.632 | 0.75 | 0.667 |
| gene | 0.333 | 1 | 0.5 | 0.125 | 0.182 |
| university | 0.5 | 1 | 0.667 | 0.0512 | 0.093 |
| food | 0* | 0* | n/a* | 0* | n/a* |
| language | 1 | 1 | 1 | 0.045 | 0.0861 |
| spacecraft | 1 | 1 | 1 | 1 | 1 |
| conference | 1 | 1 | 1 | 0.125 | 0.2222 |
| astronaut | 1 | 1 | 1 | 1 | 1 |
| composer | 0.25 | 1 | 0.4 | 0.5 | 0.3333 |
| tv program | 0* | 0* | n/a* | 0* | n/a* |
| instrument | 0* | 1* | 0* | 1* | 0* |
| recipe | 0* | 1* | 0* | 1* | 0* |
| student | 1* | 0* | 0* | 0* | 0* |
| phenotypes | 1* | 0* | 0* | 0* | 0* |
| energy | 1* | 0* | 0* | 0* | 0* |

TABLE IV
EVALUATION WITH CKAN AS BASELINE

This evaluation demonstrates that our approach is nearly as effective as the manual tagging of datasets by CKAN for dataset search.

*2) Evaluation with CKAN as baseline*

In order to better understand our automated system in comparison with the CKAN, we performed a more detailed and focused evaluation against CKAN. For this, we again utilized the twenty terms from the previous evaluation, and retrieved the search results for those twenty terms from both CKAN and our search application. By considering the CKAN results as the baseline, we calculated the Precision (P), Recall and F-measure for our search application. Here we calculated two recall values R1 and R2, where R1 considers only the 30 datasets we used for our approach and R2 considers all the datasets.

Table IV summarizes the results. Some of the extremal

| Term | Dataset |
|---|---|
| music | BBCMusic, DBTuneClassic, BBCProgram, LinkedMovieDB |
| artist | DBTuneClassic, LinkedMovieDB, BBCMusic, BBCProgram |
| biology | GeoSpecies, BBCWildLife, GeneBank, Diseasome, DrugBank |
| animal | BBCWildLife |
| geology | OrdanaceSurvey, Climb Data |
| drug | DrugBank, DailyMed, Diseasome, SIDER, MediCare |
| gene | GenBank, Diseasome, DrugBank |
| university | VIVOIndiana, eumida, ECSRKBExplorer |
| food | Foodalista |
| language | lingvoj |
| spacecraft | NASA Space Flight and Astronaut data |
| conference | Semantic Web Dog Food |
| tv program | BBC Program, BBC Music |
| instrument | BBCProgram, BBCMusic, DBTuneClassic |
| astronaut | NASA Space Flight and Astronaut data |
| composer | BBCMusic, BBCProgram, DBtuneClassic, LinkedMovieDB |
| recipe | Foodalista |
| phenotypes | Diseasome |
| student | VIVO Indiana, eumida, ECSRKBExplorer |
| energy | Linked Clean Energy Data |

TABLE V
MANUALLY CLASSIFIED DATASETS

| | CKAN | | | Our Approach | | |
|---|---|---|---|---|---|---|
| **Term** | **P** | **R** | **F** | **P** | **R** | **F** |
| music | 1 | 0.5 | 0.667 | 0.571 | 1 | 0.727 |
| artist | 1 | 0.25 | 0.4 | 0.8 | 1 | 0.9 |
| biology | 1 | 0.2 | 0.333 | 0.625 | 1 | 0.769 |
| animal | 0 | 0 | n/a | 0.333 | 1 | 0.5 |
| geology | 0 | 0 | n/a | 1 | 0.5 | 0.667 |
| drug | 1 | 0.6 | 0.75 | 1 | 1 | 1 |
| gene | 1 | 0.333 | 0.5 | 1 | 1 | 1 |
| university | 0.5 | 0.667 | 0.572 | 0.6 | 1 | 0.75 |
| food | 0 | 0 | n/a | 0.25 | 1 | 0.4 |
| language | 1 | 1 | 1 | 1 | 1 | 1 |
| spacecraft | 1 | 1 | 1 | 1 | 1 | 1 |
| conference | 1 | 1 | 1 | 1 | 1 | 1 |
| tv program | 0 | 0 | n/a | 1 | 1 | 1 |
| instrument | 1 | 0 | 0 | 0.75 | 1 | 0.857 |
| astronaut | 1 | 1 | 1 | 1 | 1 | 1 |
| composer | 1 | 0.25 | 0.4 | 1 | 1 | 1 |
| recipe | 1 | 0 | 0 | 1 | 1 | 1 |
| phenotypes | 1 | 1 | 1 | 1 | 0 | 0 |
| student | 1 | 0.5 | 0.667 | 1 | 0 | 0 |
| energy | 1 | 0.333 | 0.5 | 1 | 0 | 0 |
| Mean | 0.775 | 0.432 | 0.489 | 0.846 | 0.825 | 0.728 |

TABLE VI
COMPARISON OF OUR APPROACH AND CKAN WITH A MANUAL CURATED GOLD STANDARD

values can be explained as follows (marked by a * in the table): (1) Our search application did not return any results for the terms 'student', 'phenotypes' and 'energy'.[19] (2) CKAN did not return any results for 'instrument' and 'recipe'.[20] (3) There is no overlap between results returned by our system and CKAN for the terms 'animal', 'geology', 'food' and 'tv program'.

The results demonstrate that our approach is able to provide high recall for some terms like 'music', 'language' and 'spacecraft'. The poor precision and recall values for some terms can be due to (i) inaccuracies within CKAN which we consider as baseline here, or (ii) shortcomings in our system. We further investigate this issue in Section IV-B3 below. We also believe that our results could be improved further by increasing the number of datasets utilized by our system for generating the results.

*3) Comparison of CKAN and our approach against a manually curated gold standard*

Although CKAN is manually populated, it does have omissions and contain erroneous values. For an example, when we search for the term 'food', CKAN gives 'Semantic Web Dog Food' as a relevant result even though it is obvious that this dataset has nothing to do with 'food' as such. Hence, in order to perform a fair evaluation and to establish a baseline for our system and any future applications in the same spirit, we asked 3 different human graders to manually assign the datasets to the given terms. The output is presented in Table V. We did this, on the one hand, to achieve higher quality results which have been manually verified. On the other hand, since CKAN utilizes keyword based indexing, it affects the results obtained by using its search interface, as explained earlier for the Semantic Web Dog Food example.

We then compared our system and CKAN with this manually curated gold standard, the results are presented in Table VI. We use P for Precision, R for Recall, and F for F-Measure. Table VI shows that our search application provides nearly 90% better recall with respect to the manually verified standard, while being at par with CKAN in terms of precision.

To summarize, our approach can be helpful for systematically categorizing and finding relevant datasets from LOD. Our evaluations demonstrate that our approach provides significantly better precision and recall in retrieving LOD datasets, compared to other approaches. It also demonstrates that the state of the art of LOD searching systems fails to provide the support required for searching and retrieving relevant datasets from the LOD cloud. The reasons for the superior performance of our system lies in the utilization of a diverse classification hierarchy such as Freebase in comparison to approaches which utilize traditional indexing and manual tagging based approaches. In addition, our system is automated, and thus scales well compared to manual approaches such as the tagging used in CKAN.

## V. RELATED WORK

To the best of our knowledge this is the first effort towards automatic domain identification for LOD datasets. As we have pointed out in the paper domain identification of datasets will help to improve identification of relevant datasets. CKAN and LODStats are the state of the art for finding relevant datasets on LOD. CKAN encourages data publishers to tag their datasets with a set of predefined labels, which are then manually reviewed by the CKAN administrators. CKAN is

[19]We have listed a precision of 1 in this case, indicating that we did not have any false-positives. The precision could also be considered *undefined* in this case.

[20]We have listed a recall of 1 in this case, indicating that we did not have any true-negatives. The recall could also be considered *undefined* in this case.

used to generate the LOD bubble diagram and it provides a search interface based on the metadata provided, assigned tags and keywords. LODStats [3] is a stream based approach for gathering statistics about the datasets and it allows to search datasets based on keywords. Both CKAN and LODStats rely on the metadata provided by data publishers and hence rely on manually categorizing and describing the datasets. While this may lead to high quality descriptions, it may also result in incomplete ones, as the process is tedious and time consuming and consequently different data providers may provide uneven descriptions or metadata. For enriching metadata about the datasets, in [7] authors have presented a system to create such metadata via annotation tools and a faceted search. However, even this approach involves that the data publishers or some third party provide the annotations. In addition to these systems, semantic search engines such as Sindice, Watson and Swoogle facilitate searching for entities but none of these systems are designed specifically for dataset search.

Dataset selection and identification discussed in the context of federated querying and data interlinking. In SchemEX [8], authors provide a scalable approach for indexing LOD datasets. It provides an index by leveraging type and property information of RDF instances. In [9] authors have proposed an index structure to store dataset summaries using QTree to identify relevant data sources. These data summaries are obtained by applying a hash function to the triples of the dataset and mapped to the numerical space. In [10] the authors have used voID descriptions [11], containing metadata about datasets, to build an index which can be incorporated in query processing to determine the relevant dataset for querying. In [12], [13], [14], [15] authors have proposed different techniques for dataset identification for query answering. [16] presents an approach to identify relevant data sources for interlinking for a given particular dataset by using a semantic web index like sig.ma [17]. We believe that these approaches could also benefit from automatic topic identification for datasets, in order to reduce their search space and also to further identify more relevant results.

Another related body of work is *topic modeling*, which is about the identification of abstract topics (related clusters of words) that occur in a collection of documents. Latent Semantic Analysis [18] is a dimensionality reduction technique to identify the latent concepts which result in documents with similar topical content to be close to one another. But these latent concepts cannot be readily mapped into natural concepts. Subsequently, probabilistic approaches such as the pLSI model [19] and LDA [20] were used for topics models. Along these lines, Explicit Semantic Analysis (ESA) [21] has been proposed to use machine learning techniques together with Wikipedia as a knowledge base, to augment key word based representations with concepts from Wikipedia. Another body of related work is in the area of document/text classification into pre-defined topic hierarchies or taxonomies using machine learning techniques, such as [22] and [23]. All these systems have the advantage of text being available in the documents for classification, but in our case we only have one label for each typed instance in the dataset. A number of these systems utilize training data whereas our approach does not utilizes any training data at all.

## VI. Conclusion and Future Work

We have presented a solution for systematically identifying topic domains of LOD datasets. We have evaluated our approach against existing others and reported on a user study. Our evaluation shows the effectiveness of our approach and that it can be very useful for the LOD community in number of ways. This work has the potential to be a basis for creating a search catalogue for LOD datasets as we have shown in our evaluation. Furthermore, our work is also potentially useful for identifying datasets for the purpose of interlinking.

Our approach currently draws some of its strength from the richness of Freebase. However, only the first *Category Identification* step described in Section II actually depends on Freebase, the remainder of the approach is completely generic. Returning to the description of the Category Identification step in Section II, note that the only thing we need is a way to assign both a general *domain* and a more specific *type* to each instance in a dataset. Several alternatives suggest how to approach this, some of which will again reuse LOD datasets. Complementing the use of Freebase with other appropriate knowledge sources is not only interesting in order to improve performance or topic coverage of our system. It is also needed for full-scale topic domain identification for all LOD datasets, as the full use of Freebase is restricted daily basis. We intend to work on such alternatives.

We are confident that the LOD community can benefit from our approach. Our work can in principle easily be integrated with LOD meta data repositories such as CKAN, LOD Stats and Sindice, to allow people to gain a better understanding of the datasets. CKAN can use this for topic identification as an alternative or replacement for the manual assignment of topics. Furthermore, the LOD Bubble Diagram could be organized in a better way with improved topic domain identifications. We plan to extend our coverage beyond the 30 datasets presented in the paper and to provide a comprehensive coverage of LOD datasets.

*Acknowledgement.* This work was supported by the National Science Foundation under award 1143717 “III: EAGER Expressive Scalable Querying over Linked Open Data.” Pascal Hitzler acknowledges support by the National Science Foundation under award 1017225 “III: Small: TROn – Tractable Reasoning with Ontologies.” Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## References

[1] D. A. Ferrucci, E. W. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. M. Prager, N. Schlaefer, and C. A. Welty, “Building Watson: An Overview of the DeepQA project,” *AI Magazine*, vol. 31, no. 3, pp. 59–79, 2010.

[2] P. Jain, P. Hitzler, A. P. Sheth, K. Verma, and P. Z. Yeh, “Ontology Alignment for Linked Open Data,” in *Proceedings of the 9th International*

*Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010*, vol. 6496. Springer-Verlag, 2010, pp. 402–417.

[3] 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, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, vol. 7603. Springer, 2012, pp. 353–362.

[4] G. Tummarello, R. Delbru, and E. Oren, "Sindice.com: Weaving the open linked data," in *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007*, vol. 4825. Springer, 2007, pp. 552–565.

[5] M. d'Aquin and E. Motta, "Watson, more than a Semantic Web search engine," *Semantic Web*, vol. 2, no. 1, pp. 55–63, 2011.

[6] T. W. Finin, L. Ding, R. Pan, A. Joshi, P. Kolari, A. Java, and Y. Peng, "Swoogle: Searching for Knowledge on the Semantic Web," in *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*. AAAI Press / The MIT Press, 2005, pp. 1682–1683.

[7] M. Frosterus, E. Hyvönen, and J. Laitio, "Datafinland - A Semantic Portal for Open and Linked Datasets," in *The Semanic Web: Research and Applications - 8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece, May 29 - June 2, 2011, Proceedings, Part II*, vol. 6644. Springer, 2011, pp. 243–254.

[8] M. Konrath, T. Gottron, S. Staab, and A. Scherp, "Schemex - Efficient construction of a data catalogue by stream-based indexing of linked data," *J. Web Sem.*, vol. 16, pp. 52–58, 2012.

[9] A. Harth, K. Hose, M. Karnstedt, A. Polleres, K.-U. Sattler, and J. Umbrich, "Data summaries for on-demand queries over linked data," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 411–420.

[10] O. Görlitz and S. Staab, "SPLENDID: SPARQL Endpoint Federation Exploiting VOID Descriptions," in *Proceedings of the Second International Workshop on Consuming Linked Data (COLD2011), Bonn, Germany, October 23, 2011*, vol. 782. Bonn, Germany: CEUR-WS.org, 2011.

[11] K. Alexander, R. Cyganiak, M. Hausenblas, and J. Zhao, "Describing Linked Datasets - On the Design and Usage of voID, the "vocabulary of interlinked datasets"," in *Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW 2009, Madrid, Spain, April 20, 2009*, vol. 538. CEUR-WS.org, 2009.

[12] A. Schwarte, P. Haase, K. Hose, R. Schenkel, and M. Schmidt, "FedX: Optimization Techniques for Federated Query Processing on Linked Data," in *Proceedings of the 10th international conference on The semantic web - Volume Part I*, ser. ISWC'11, vol. 538. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 601–616.

[13] O. Hartig, C. Bizer, and J. C. Freytag, "Executing SPARQL Queries over the Web of Linked Data," in *The Semantic Web - ISWC 2009, 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings*, vol. 5823. Springer, 2009, pp. 293–309.

[14] H. R. de Oliveira, A. T. Tavares, and B. F. Lóscio, "Feedback-based data set recommendation for building linked data applications," in *I-SEMANTICS 2012 - 8th International Conference on Semantic Systems, I-SEMANTICS '12, Graz, Austria, September 5-7, 2012*. ACM, 2012, pp. 49–55.

[15] T. Tran, L. Zhang, and R. Studer, "Summary Models for Routing Keywords to Linked Data Sources," in *The Semantic Web - ISWC 2010 - 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part I*, vol. 6496. Springer, 2010, pp. 781–797.

[16] A. Nikolov, M. d'Aquin, and E. Motta, "What Should I Link to? Identifying Relevant Sources and Classes for Data Linking," in *The Semantic Web - Joint International Semantic Technology Conference, JIST 2011, Hangzhou, China, December 4-7, 2011. Proceedings*, vol. 7185. Springer, 2011, pp. 284–299.

[17] G. Tummarello, R. Cyganiak, M. Catasta, S. Danielczyk, R. Delbru, and S. Decker, "Sig.ma: Live views on the Web of Data," in *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*. ACM, 2010, pp. 1301–1304.

[18] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[19] T. Hofmann, "Probabilistic Latent Semantic Analysis," in *UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30 - August 1, 1999*. Morgan Kaufmann, 1999, pp. 289–296.

[20] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, Mar. 2003.

[21] E. Gabrilovich and S. Markovitch, "Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis," in *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, vol. 6, 2007, pp. 1606–1611.

[22] L. Cai and T. Hofmann, "Hierarchical Document Categorization with Support Vector Machines," in *Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, Washington, DC, USA, November 8-13, 2004*. ACM, 2004, pp. 78–87.

[23] P.-Y. Hao, J.-H. Chiang, and Y.-K. Tu, "Hierarchically svm classification based on support vector clustering method and its application to document categorization," *Expert Syst. Appl.*, vol. 33, no. 3, pp. 627–635, Oct. 2007.