The VLDB Journal (2020) 29:251–272
https://doi.org/10.1007/s00778-019-00564-x

SPECIAL ISSUE PAPER

Dataset search: a survey

Adriane Chapman[1] · Elena Simperl[1] · Laura Koesten[1] · George Konstantinidis[1] · Luis-Daniel Ibáñez[1] · Emilia Kacprzak[2] · Paul Groth[3]

Received: 27 December 2018 / Revised: 15 July 2019 / Accepted: 12 August 2019 / Published online: 24 August 2019

Abstract
Generating value from data requires the ability to find, access and make sense of datasets. There are many efforts underway to encourage data sharing and reuse, from scientific publishers asking authors to submit data alongside manuscripts to data marketplaces, open data portals and data communities. Google recently beta-released a search service for datasets, which allows users to discover data stored in various online repositories via keyword queries. These developments foreshadow an emerging research field around dataset search or retrieval that broadly encompasses frameworks, methods and tools that help match a user data need against a collection of datasets. Here, we survey the state of the art of research and commercial systems and discuss what makes dataset search a field in its own right, with unique challenges and open questions. We look at approaches and implementations from related areas dataset search is drawing upon, including information retrieval, databases, entity-centric and tabular search in order to identify possible paths to tackle these questions as well as immediate next steps that will take the field forward.

Keywords Dataset search · Dataset retrieval · Dataset · Information search and retrieval

1 Introduction

Data is increasingly used in decision making: to design public policies, identify customer needs, or run scientific experiments [64,173]. For instance, the integration of data from deployed sensor systems such as mobile phone networks, camera networks in intelligent transportation systems (ITS) [103] and smart meters [3] is powering a number of innovative solutions, such as the city of London's oversight dashboard [17]. Datasets are increasingly being exposed for trade within data markets [13,70] or shared via open data portals [41,80,97,125,144,174] and scientific repositories [5,57]. Communities such as Wikidata or the Linked Open Data Cloud [125] come together to create and maintain vast, general-purpose data resources, which can be used by developers in applications as diverse as intelligent assistants, recommender systems and search engine optimization. The common intent is to broaden the use and impact of the millions of datasets that are being made available and shared across organizations [24,148,184]. This trend is reinforced by advances in machine learning and artificial intelligence, which rely on data to train, validate and enhance their algorithms [159]. In order to support these uses, we must be able to search for datasets. Searching for data in principled ways has been researched for decades [42]. However, many properties of *datasets* are unique, with interesting requirements and constraints, which have been recognized by the recent release of Google Dataset Search [141]. There are many open problems across dataset search, which the database community can assist with.

✉ Adriane Chapman
adriane.chapman@soton.ac.uk

Elena Simperl
E.Simperl@soton.ac.uk

Laura Koesten
L.M.Koesten@soton.ac.uk

George Konstantinidis
G.Konstantinidis@soton.ac.uk

Luis-Daniel Ibáñez
L.D.Ibanez@soton.ac.uk

Emilia Kacprzak
emilia.kacprzak@theodi.org

Paul Groth
p.groth@uva.nl

[1] University of Southampton, Southampton, UK

[2] The Open Data Institute, London, UK

[3] University of Amsterdam, Amsterdam, The Netherlands

Currently, there is a disconnect between what datasets are available, what dataset a user needs, and what datasets a user can actually find, trust and is able to use [24,159,167]. Dataset search is largely keyword-based over published metadata, whether it is performed over web crawls [66,161] or within organizational holdings [80,97,171]. There are several problems with this approach. Available metadata may not encompass the actual information a user needs to assess whether the dataset is fit for a given task [106]. Search results are returned to the user based on filters and experiences that worked for web-based information, but do not always transfer well to datasets [68]. These limitations impact the use of the retrieved data—machine learning can be unduly affected by the processing that was performed over a dataset prior to its release [168], while knowing the original purpose for collecting the data aids interpretation and analysis [185]. In other words, in a dataset search context, approaches need to consider additional aspects such as data provenance [27,67,81,112,135,187], annotations [86,124,189], quality [155,175,195], granularity of content [105], and schema [4,18] to effectively evaluate a dataset's fitness for a particular use. The user does not have the ability to introspect over large amounts of data, and their attention must be prioritized [11]. In some cases, a user's need may even require integrating data from different sources to form a new dataset [63,155]. Furthermore, using data is sometimes constrained by licenses and terms and conditions, which may prohibit such integration, especially when personal data is involved [136].

In order to realize the full potential of the datasets we are generating, maintaining and releasing, there is more research that must be done. Dataset search has not emerged in isolation, but has built on foundational work from other related areas. In Sect. 2, we outline the basic dataset search problem and briefly review these areas. Current commercial dataset search offerings are introduced in Sect. 3, while Sect. 4 provides an overview of ongoing dataset search research. Finally, Sect. 5 discusses several open problems, while Sect. 6 highlights a possible route to take steps to advance the field.

2 Background

To understand the fundamental problem of dataset search, we define a dataset. The concept of dataset is abstract, admitting several definitions depending on the particular community [24,148]. There is a large body of work discussing the nature of data and its relation to practice and reuse [24,25]. For example, the statistical data and metadata exchange initiative (SDMX) [162] defines a dataset as 'a collection of related observations, organized according to a predefined structure'. This definition is shared by the DataCube working group at the World Wide Web Consortium (W3C), which adds the notion of a 'common dimensional structure' [179]. Meanwhile, the Organization for Economic Co-operation and Development (OECD), citing the US bureau and census, uses 'any permanently stored collection of information usually containing either case level data, aggregation of case level data, or statistical manipulations of either the case level or aggregated survey data, for multiple survey instances' [162]. The Data Catalog Vocabulary, another W3C effort, [127] includes a dataset class, defined as a 'collection of data, published or curated by a single agent, and available for access or download in one or more formats.' Finally, for the MELODA (MEtric for reLeasing Open DAta) initiative, a dataset is a 'group of structured data retrievable in a link or single instruction as a whole to a single entity, with updating frequency larger than a once a minute' [131]. Building upon these proposals, for the purposes of this paper, we will use the following definition:

Definition 1 Dataset: A collection of related observations organized and formatted for a particular purpose.

A dataset can be a set of images, or graphs, or documents, as well as the classic table of data. Thus, dataset search involves the discovery, exploration, and return of datasets to an end-user. However, within this work, we focus on alphanumeric data (e.g., text, entities, data). While datasets may also comprise images, or graphs, search techniques for these modalities contain both alphanumeric search techniques for metadata and specialized techniques based on the structure of the data. Thus, to be more general in this survey, we discuss techniques for alphanumeric data. We note two very distinct types of dataset search in this work. In what we will call 'basic' dataset search, the set of related observations are organized for a particular purpose and then released for consumption and reuse. We see this pattern of interaction within individual data repositories, such as for research data (e.g., Figshare [171], Dataverse [5], Elsevier Data Search [57]), open data portals [41,80,97,125,144,174] and search engines such as DataMed [161] and Google Dataset Search [141]. A basic search, using any of these services, is discussed in Example 1. Alternatively, a dataset search may involve a set of related observations that are organized for a particular purpose by the searcher themselves. This pattern of behavior is particularly marked in data lakes [62,156], data markets [13,70], and tabular search [114,198]. Example 2 illustrates this kind of data search.

Example 1 *(Basic dataset search)* Imagine you want to write an article on how Hurricane Sandy impacted the gasoline prices in New York City in the week after the incident. Consider the two datasets shown in Fig. 1. Dataset *A* is from the American Automobile Association (AAA) and dataset *B* is from Twitter. Both document the gasoline available for purchase in New York City in the week after Hurricane Sandy. The choice of which dataset to use depends on

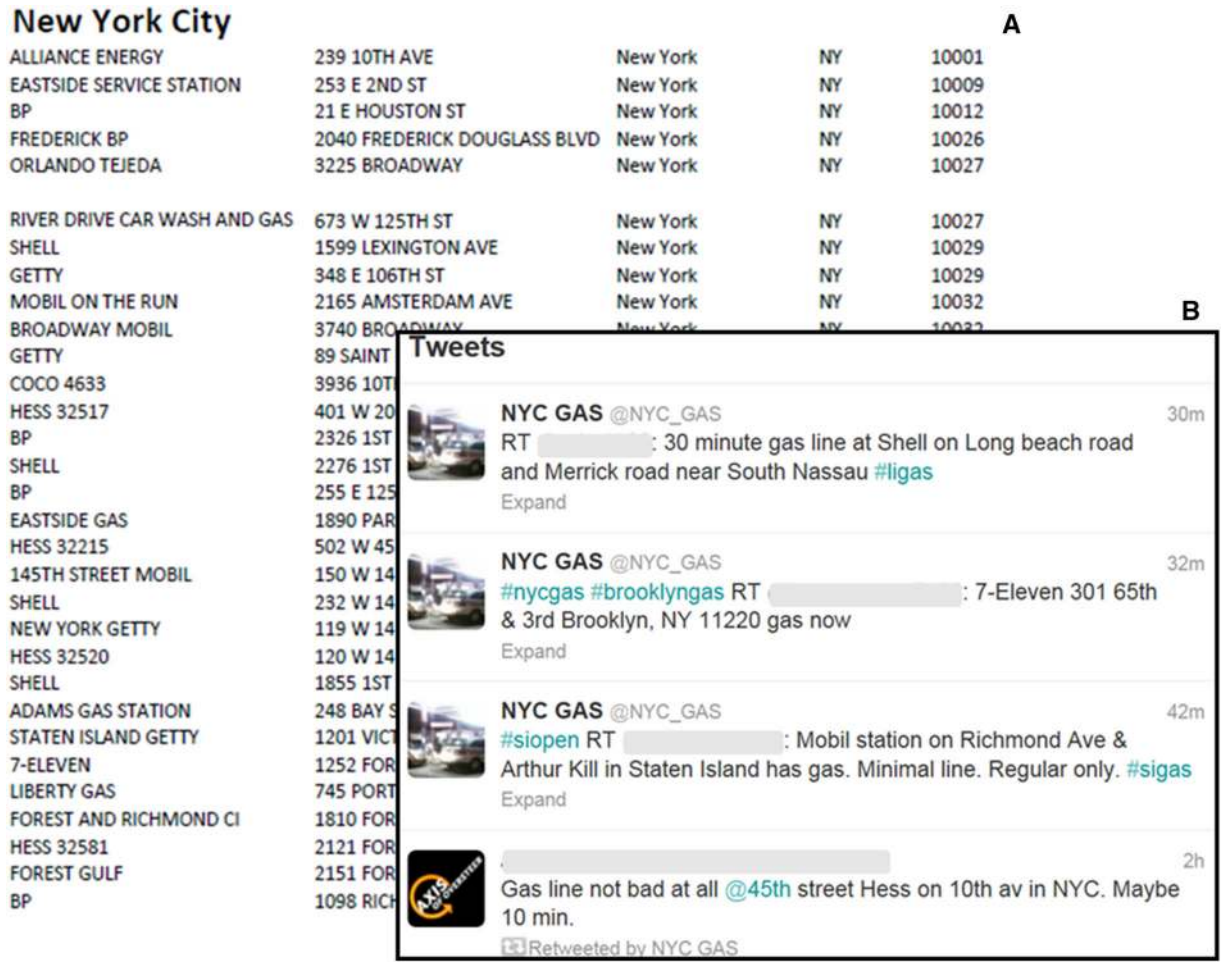

Fig. 1 Datasets about gasoline availability in New York City in the week after Hurricane Sandy in 2012. **a** The American Automobile Association (AAA) created a structured dataset twice post-Sandy by phoning every gas station in the NYC area. It was complete, easy to use (CSV), accurate, clean, and out of date by the time it was released. **b** The second dataset is a collection of tweets to NYC_GAS. It was incomplete, required natural language processing (NLP) techniques to use, was dirty with respect to place names and addresses, but up to date and timely throughout post-hurricane clean-up efforts

the specifics of the information need, potentially the purpose and requirements of algorithms or processing methods, as well as the user's tool-set and data literacy. In order to find the right dataset, a user must issue a query that will return datasets, not tuples, documents or corpora. Differences inherent in the datasets should alter their ranking. For instance, a user who requires easy-to-use data, with fewer restrictions on timeliness, may feel that the AAA dataset is a better fit than the other one. A user who wishes to establish an accurate timeline of gas in NYC would have a different assessment. These two scenarios have different requirements and therefore would assess the datasets differently. Moreover, both users start filtering results based on content (gasoline), but use very different criteria and metrics to rank the datasets.

Example 2 *(Constructive dataset search)* The Centro De Operacoes Prefeitura Do Rio in Rio de Janeiro, Brazil, is developing a strategy to prevent and manage floods in the city. The city planners follow a data-informed approach, where they mash up several data sources, including traffic and public transport; utility and emergency services; weather; and citizen reports [103]. Consider a simple scenario in which datasets on weather, highlighting rain amounts that could trigger a flash flood, are integrated on the fly with datasets on traffic volume and augmented with identification of emergency response services in order to create a dataset that highlights the current populations at risk during an event. A recent extension to RapidMiner highlights the opportunities inherent in creating such as dataset, with additional examples [63].

2.1 Overview of dataset search

Figure 2 contains a high-level view of the search process and mappings of the main process steps to topics researched in related communities.

A general approach to providing search over datasets is to model the user experience over existing keyword-based information retrieval search systems, where a user poses a query and a ranked list of existing datasets is returned.

Querying. In dataset search, a query is typically a keyword or Contextual Query Language (CQL) expression [163]. Figure 3 shows the search interface for the UK government's open data portal [174]. In addition to the keywords search box, the "Filter by" boxes allow the user to subset the data according to predefined categories. As we discuss search techniques from several different disciplines below, we use the term 'query' to mean a semantically and syntactically correct expression of search for that specific technology. For instance, within a database, a query would be expressed in SQL, while in information retrieval, a query would be expressed in CQL.

Query handling. The information submitted by the user is used to search over the metadata published about a dataset. Results are produced based on how similar the metadata is to the search terms.

Data handling. Publishers populate the metadata about their dataset, including title, description, language, temporal coverage, etc. They can use vocabularies such as DCAT [127], schema.org [71] or CSV on the Web [170] as a starting point. The goal of these vocabularies is to provide a uniform way of ensuring consistency of data types and formats (e.g., uniqueness of values within a single column) for every file, which can provide basis for validation and prevent potential errors. Publishers sometimes add descriptions to their datasets to aid sensemaking [105,140,189]. Either way, this step is mostly manual and hence resource-intensive, which means that more often than not dataset descriptions are incomplete or do not contain enough detail. This limits the capabilities of query handling methods, which attempt to match search terms to the descriptions.

Results presentation. Search Engine's Results Pages (SERPs) for dataset search currently follow a traditional *10 blue links* paradigm, as can be seen on many data portals [5,80,97,174]. Basic filtering options, as shown in Fig. 3, are sometimes available for faceted search. Clicking on a search result usually takes the user to a preview page that contains metadata, a free-text summary, and sometimes a sample or visualization of the data. While Google Dataset Search [66] also follows a traditional result presentation as a list, they display a split interface. This presents a large number of search results for scrolling on the left side and a reduced version of a dataset preview page with links to one (or multiple) repositories that hold the respective dataset on the right side.

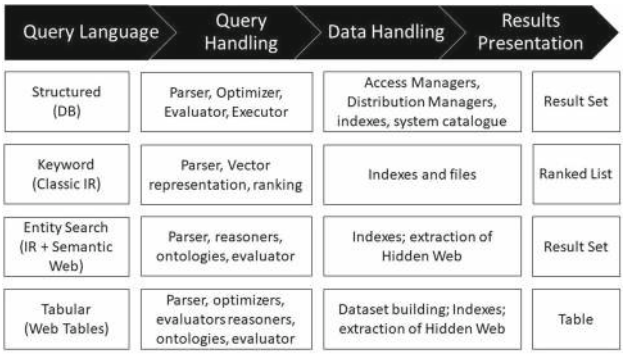

Fig. 2 An abstract view of the search process, comprising of querying, query processing, data handling and results presentation, alongside approaches to each step by different related communities

Fig. 3 Dataset search engine for the UK government's open data portal, data.gov.uk. Form inputs create CQL statements to query the underlying data

2.2 Common search architectures

Searches for datasets can be local, e.g., within a single repository [5,57,156,171] or global. In a similar manner to a distributed database, given a query Q and a set of datasets (the *sources*), the query engine first selects the datasets relevant to the query [160,177] and then chooses between different approaches: aggregating the datasets locally, using distributed processing as in Hadoop [188], or query federation [143].

The dataset search problem can be addressed at various levels. Services such as Google Dataset Search [141] and DataMed [161] crawl across the web and facilitate a global search across all distributed resources. These approaches use tags found in metadata mark-up, expressed in vocabulary terms from schema.org [71] or DCAT [127], to structure and identify the metadata considered important for datasets.

However, the problem also exists at a local level, including open government data portals such as data.gov.uk [174], organizational data lakes [156], scientific repositories such as Elsevier's [57] and data markets [13,70]. Across all these systems, users are attempting to discover and assess datasets for a particular purpose. Supporting them requires frameworks, methods and tools that specifically target data as its input form and consider the specific information needs of data professionals.

2.3 Other search sub-communities

Search has been addressed in a range of scenarios, depending on the types of data and methods used. Relevant subdisciplines include databases, document search (classic information retrieval), entity-centric search (tackled in the context of the semantic web, knowledge discovery in databases and information retrieval), and tabular search (which draws upon methods from broader data management, IR and sometimes entity-centric search).

Figure 2 lists some of the most important methods used in these sub-disciplines to implement core search capabilities from query writing and handling to results presentation. While dataset search is a field in its own right, with distinct challenges and characteristics, it shares commonalities and draws upon insights from all these disciplines. In this section, we provide a very brief review of the focus and tools each community uses. We focus specifically on those in which the type of object returned is the same as the underlying data, e.g., a result set of data from a database of data, or a document from a corpus of documents. We neglect approaches such as question answering [111], which involve additional processing and reasoning steps.

2.3.1 Databases

The classic pipeline for search within a database begins with a structured query, followed by parsing the query [38,117,121]; creating an evaluation plan [116]; optimizing the plan [37,87]; and executing the plan utilizing appropriate indexes and catalogues [19].

In addition to the classic database search pipeline, we wish to draw attention to recent work to uncover more data from hidden areas of the web: Hidden/Deep web search.

Hidden/deep web search. The *hidden*, or *deep*, web refers to content that lies "behind" web forms typically written in HTML [28,29,77,100,128], and ranging from medical research data to financial information and shopping catalogues. It has been estimated that the amount of content available in this way is an order of magnitude larger than the so-called *surface* web, which is directly accessible to web crawlers [77,128].

There have been two main approaches to searching for data on the deep web. The first uses more traditional techniques to build vertical search engines, whereby semantic mappings are constructed between each website and a centralized mediator tailored to a particular domain. Structured queries are posed on the mediator and redirected over the web forms using the mappings. Kosmix [154] (later transformed into WalmartLabs.com) was such a system presenting vertical engines for a large number of domains, ranging from health, and scientific data to car and flight sales. Sometimes systems learn the forms' possible inputs, and create centralized mediated forms [77].

A second group of approaches tries to generate the resulting web pages, usually in HTML, that come out of web form searches. Google has proposed a method for such *surfacing* of deep web content by automatically estimating input to several millions of HTML forms, written in many languages and spanning over hundreds of domains, and adding the resulting HTML pages into its search engine index [128]. The form inputs are stored as part of the indexed URL. When a user clicks on a search result, they are directed to the result of the (freshly submitted) form.

2.3.2 Information retrieval

Several classes of IR systems existing, including document search, web search and engines for other types of objects (e.g., images, people etc.). When working on text, IR uses a range of statistical and semantic techniques to compute the relevance of search terms of documents. Specialized search engines are tailored to the characteristics of the underlying resources. For example, email search considers aspects such as sender and receiver addresses, topic or timestamp to define relevance functions [2]. Due to their specificity and limited scope of resources, vertical search engines often offer greater precision, utilize more complex schemas to match searching scenarios, and tend to support more complex user tasks [120,180,194].

2.3.3 Entity-centric search

In entity-centric search information is organized and accessed via entities of interest, and their attributes and relationships [15]. A comprehensive overview of the area is available in [14]. The problem has been tackled mostly by the semantic web and knowledge discovery communities.

From a semantic web perspective, efforts have been directed toward creating machine-understandable graph-based representations of data [79]. Researchers have proposed languages, models and techniques to publish structured data about entities and link entities to each other to facilitate search and exploration in a decentralized space, replicating search and browsing online. The World Wide Web Con-

sortium (W3C) settled on the Resource Description Format (RDF) as a standard model for representing and exchanging data about resources, which can refer to conventional web content (information resources), as well as entities in the offline world such as people, places and organizations (non-information resources). Both are identified by *International Resource Identifiers* (IRIs). Properties link entities or attach attributes to them. By reusing and linking IRIs, publishers signal that they hold data about the same entity, therefore enabling queries across multiple datasets without any additional integration effort.

To take advantage of these features, data needs to be encoded and published as *linked data* [9], which refers to a set of technologies and architectures including IRIs, RDF, RDFS (RDF Schema) and HTTP. The structure of the data is defined in vocabularies, which can be reused across datasets to facilitate data interpretation and interlinking. Platforms such as Linked Open Vocabulary portal[1] assist publishers looking for a vocabulary for their data with search and exploration capabilities over hundreds of vocabularies developed by different parties.

Interlinking is concerned with two related problems. First is *entity resolution*: given two or more datasets, identify which entities and properties are the same. A general framework of entity resolution is described [40]. It covers the design of *similarity metrics* to compare entity descriptions, and the development of *blocking* techniques to group roughly similar entities together to make the process more efficient. More recent efforts have proposed *iterative* approaches, where discovered matches are used as input for computing similarities between further entities. The second part of interlinking is referred to as *link discovery*, where given two datasets, one has to find properties that hold between their entities. Properties can be equivalence or equality, as in entity resolution, or domain specific such as 'part-of' [79].

Linked data facilitates entity-centric search. A user can express a query using a structured language such as SPARQL, which includes entities, entity classes, as well as properties and values. Queries are translated into an RDF graph that is matched against the published data, which is also available as RDF [197,199]. Similar to before, queries can be answered locally, against an RDF data store, or globally, using a range of techniques.

From a knowledge discovery perspective, significant efforts have been directed toward the construction of knowledge graphs (KGs), which are large collections of interconnected entities, which may or may not be encoded using linked data. Building KGs requires a range of capabilities, from vocabularies to describe the domain of interest to extraction algorithms to take data from different sources and map it to graphs, to curation and evolution. Knowledge discovery shares many methods and challenges with the semantic web, the main difference being that the former focuses on building a (centralized) graph, which enhances the results of data mining and machine learning processes [23,54,166], while the latter is about managing information in open, decentralized settings such as the web.

[1] https://lov.linkeddata.es.

2.3.4 Tabular search

In tabular search, users are interested in accessing data stored in one or more tables. The overall aim is to discover specific pieces of information, such as attribute names or extend tables with new attributes. [190] identified three core tasks in this context:

1. Augmentation by attribute name—given a populated table and a new column name (i.e., attribute), populate the column with values. This is also referred to *table extension* elsewhere [28]. One can see this as finding tables which can be joined.
2. Attribute discovery—given a populated table, discover new potential column names.
3. Augmentation by example—given a populated table where some values are missing, fill in the missing values. This is often referred to as *table completion* in the literature [197] and resembles finding tables which can be unioned.

It is important to distinguish between *table* and *tabular* search. Table search is a sub-task of dataset search, where the user issues a keyword query and the result is available as tables, for example in CSV format. Tabular search is about engaging with one or more tables with the aim to manipulate and extend them. Information needs, for instance to discover attributes and tables to extend or complete, are expressed as tables. One of the challenges in tabular search is to answer the latent information need of the user.

Table extension. [115] distinguishes between constrained and unconstrained table extension. Constrained table extension is essentially the augmentation by previously defined attribute names. Unconstrained table extension also involves the addition of new columns to a table, but with no predefined label for the attribute. One can think of this as attribute discovery followed by constrained table extension.

A common technique to perform table extension is to discover existing tables through table similarity—in particular by measuring *schema similarity* [51]. In fact, table extension was introduced by [28] where they defined a special operator EXTEND that would discover similar web tables to the given input table. Similarity here is computed with respect to the schema of the table. The values of the most similar table are then used to populate the input table's addi-

Table 1 Search technology used across implementations

	Entity-centric	Information retrieval	Database	Tabular search
Open data publishing platforms (e.g., CKAN, Socrata)		•		
Data marketplaces	•			
Linked data search engines	•	•	•	
Google DSS	•	•		

tional column. The Infogather system [190] uses a similar approach, but instead of just calculating the direct similarity between the input table and potential augmenting tables, it also takes into the account the neighborhood around the potential augmenting tables. These indirect tables provide ancillary information that can be better suited for augmentation than the tables with the highest similarity to the input. Of interest, [51] have discovered that there seems to be a latent link structure between web tables. Recent work in table similarity has shown that semantic similarity using embedding approaches can improve performance over syntactic measures [198].

Table completion. This task also relies heavily on table similarity as the mechanism for finding potential values that can be added to a table. [197] defines the notion of *row population*, which adds new rows to a table. For simplicity, we view this as a type of table completion in which the values to be completed form an additional row. Even more broadly, one could provide a set of columns as a query and have the system fill in the remaining rows [151]. The task of table completion can be seen as entity-set completion, where the goal is to complete a list given a set of seed entities [51,197]. This task is relevant for a number of other scenarios, including entity-centric search [16] and knowledge-base completion [49]. The completion of rows is similar to the broad problem of imputation and dealing with incomplete data [132]. Specific work in the context of the web has looked at performing imputation through the use of external data [1,123,169]. Much of that work has used web tables as the data source, and hidden/deep web techniques as discussed above could be applied.

3 Current dataset search implementations

There are many functioning versions of dataset search in production today. In this section, we break down the set of dataset search services that exist according to their focus and how they deal with datasets. We distinguish between the two scenarios discussed in Examples 1 and 2 and between centralized and decentralized architectures. For the latter, the search engine needs a way to discover the datasets as well as handle user queries and present results.

Table 1 matches the implementations discussed to the technology described in Sect. 2. Note that at this time, we can find no examples of tabular search being used commercially.

The common theme of current dataset search strategies, both on the web and within the boundaries of a repository, is the reliance on dataset publishers tagging their data with appropriate information in the correct format. Because current dataset search only uses the metadata of a dataset, it is imperative that these metadata descriptions are correct and maintained. Other, domain-specific solutions function in similar ways. Especially for scientific datasets there are initiatives aiming to support the creation of better and more unified metadata, such as for instance CEDAR by the Metadata Center[2] or 'Data in Brief' submissions supported by Elsevier.[3]

In aid of better searches, there are several attempts at monitoring and working over data portals to provide a meta-analysis. For instance, the Open Data Portal Watch [138,139] currently watches 254 open data portals. Once a week, the metadata from all watched portals is fetched, the quality of the metadata computed, and the site updated to allow a cohesive search across the open data. Similarly, the European Data Portal[4] harvests metadata of public sector datasets available on public data portals across European countries, in addition to educating about open data publishing.

3.1 Basic, centralized search

3.1.1 Open government data portals

Open data portals [41,80,97,125,144,174] allow users to search over the metadata of available datasets. One of the most popular portal software is CKAN [41]. It is built using Apache Solr,[5] which uses Lucene to index the documents. In this scenario, the documents are the datasets' metadata provided by the publishers, expressed in CKAN. From a search point of view, datasets and their metadata are reg-

[2] https://metadatacenter.org/.

[3] https://www.journals.elsevier.com/data-in-brief.

[4] https://www.europeandataportal.eu.

[5] http://lucene.apache.org/solr/.

istered to the portal by their owners and there is no need to discover the datasets in the wild or come up with a common way to describe them. There are several competitors to CKAN, such as Socrata and OpenDataSoft, but from a dataset search point of view they have many similarities—they assume the datasets are available and accompanied by metadata encoded in the same way. Implemented this way, dataset search has many limitations, which are mostly due to the quality of the metadata accompanying the datasets and to the lack of appropriate capabilities to match keyword-based queries and metadata and come up with a meaningful ranking [68]. In many cases the metadata does not describe the full potential of the data, so some relevant datasets may not be presented as a result to a query simply because appropriate keywords were not used in the description.

3.1.2 Enterprise search

Proprietary data portals are not much different from an architecture point of view. In 2016, Google introduced *Goods*, an enterprise dataset search system, to manage datasets originating from different departments within the company with no unified structure or metadata [74]. In this catalog, related datasets are clustered based on the structure of the dataset or gathering frequency. Members of a group then become a single entry in the catalog. This helps to structure the catalog and also reduces the workload of metadata generation and schema computing. Within the Goods system each dataset entry has an overview of the dataset presented on a profile page. Using this profile, users can judge the dataset's usefulness to their task. Keyword queries are then laid on top of this structure, producing a ranked result list of datasets as an output. Search functionality was built based on an inverted index of a subset of the dataset's metadata. In the absence of the information on the importance of each resource, Halevy et al. [74] propose to rank the datasets based on heuristics over the type of a resource, precision of keyword match, if the dataset is used by other datasets and if the dataset contains an owner-sourced description.

3.1.3 Scientific data portals

Several commercial portals provide access to scientific datasets, including Elsevier [57], Figshare [171] and Dataverse [5]. They operate in a similar way to the other types of systems described in this section, offering keyword- or faceted search over metadata records of a centralized pool of datasets that is compiled with the help of data publishers.

3.1.4 Data marketplaces

Finally, data marketplaces exist as a way for organizations to realize value for their data [13,70]. From a search point of view, they match user queries to dataset descriptions, which may include a bespoke set of metadata attributes related to accessibility or price. The greatest challenge in this case is in finding a query handling approach that can give the user an estimate of the value of the data without computing the result.

3.2 Basic, decentralized search

3.2.1 Search over linked data

As noted in Sect. 2, linked data facilitates dataset search at web scale. This is exemplified in approaches such as [76], where new linked datasets are discovered during query execution, by following links between datasets and continuously adding RDF data to the queried dataset. There is also a large body of the literature and prototypical implementations for searching linked data in a native semantically heterogeneous and distributed environment [50,60,83,129,178], where semantic links are used to come up with an estimate of the importance of each dataset and rank search results.

3.2.2 Google Dataset Search

Following their work on Goods, in 2018 Google introduced a vertical web search engine tailored to discover datasets on the web [141]. This system uses schema.org [71] and DCAT [127]. Based on the Google Web Crawl, they crawl the web for all datasets described with the use of the schema.org `Dataset` class, as well as those describing their datasets using DCAT, and collect the associated metadata. They further link the metadata to other resources, identify replica and create an index of enriched metadata for each dataset. The metadata is reconciled to the Google knowledge graph and search capabilities are built on top of this metadata. The indexed datasets can be queried via keywords and CQL expressions [163].

3.2.3 Domain-specific search

Some search services focus on datasets from particular domains. They propose bespoke metadata schemas to describe the datasets and implement crawlers to discover them automatically. For instance, DataMed, a biomedical search engine uses a suite of tags, DATS, to allow a crawler to automatically index scientific datasets for search [161]. The Open Contracting Partnership released a Open Contracting Data Standard that identifies information needed about contracts to allow their crawler to access and catalogue contracting datasets [147].

3.3 Constructive search

Many private companies have understood that data is a commodity that can be effectively monetized. Some companies, such as Thomson Reuters, have been collecting data to create datasets for sale for decades.[6] In the same time, companies such as OpenCorporates use public data sources, with provenance, to gather information on legal entities. This dataset is then made publicly available.[7] Similarly, Researchably compiles information from scientific publications and makes interest-specific datasets for sale to biotech companies.[8] In all of these cases, the data exists in a scattered manner and the company provides value by gathering, organizing and releasing it as a constructed dataset.

Data marketplaces can offer similar services as well. Unlike the previous examples, they provide a catalog of datasets for users to purchase. While the user is able to download an entire dataset from the marketplace, it is also possible to access subsets of the data as needed to construct a new dataset.

4 Survey of dataset search research

This section surveys the current work related to dataset search. To organize it, we utilize the headings from Fig. 2, corresponding to the search process.

4.1 Querying

Creating queries. Users interact with datasets in a different manner than they interact with documents [99]. While this study is limited to social scientists, it indicates that users have a higher investment in the results and are thus willing to spend more time searching. Moreover, the relationship of the dataset to the task at hand may play a larger role in dataset search; e.g., two datasets about cars could fit within a user's ability to understand and utilize, but may have different results depending on the goal of the task.

Data-centric tasks can be categorized into two categories: (1) *Process-oriented tasks* in which data is used for something transformative, such as using data in machine learning processes; and (2) *Goal-oriented tasks* in which data is, e.g., used to answer a question [106]. While the boundaries between the two categories are somewhat fluid and the same user might engage in both types of tasks, the primary difference between them lies in the 'user information needs', i.e., the details users need to know about the data in order to interact with it effectively. For process-oriented tasks, aspects such as timeliness, licenses, updates, quality, methods of data collection and provenance have a high priority. For goal-oriented tasks, intrinsic qualities of data such as coverage and granularity play a larger role. As yet, beyond the user filtering by certain characteristics, there is no way to state the task needs in the query. There has not yet been a movement away from keywords and CQL to query datasets.

Query types. As stated earlier, most queries for datasets use keywords or CQL over the metadata of the dataset. A formal query language that supports dataset retrieval does not yet exist. Instead, specific query interfaces are created for the underlying data type, e.g., [90] provides a SQL interface over text data and [138] for temporal and spatial data. Current implementations provide platform specific faceted search to allow basic filtering for categories such as publisher, format, license or topics (for instance [174]).

4.2 Query handling

As stated in Sect. 2, most dataset searches operate over the dataset's metadata. Unfortunately, low metadata quality (or missing metadata) affects both the discovery and the consumption of the datasets within Open Data Portals [175]. The success of the search functionality depends on the publishers knowledge of the dataset and the quality of the descriptions they provide.

Moving away from just searching over the metadata, [172] use the data type and column information for mapping columns in a query to the underlying table columns, while [151] allow keyword queries over columns. Similarly, [72] describe how to map structured sources into a semantic search capability. This is taken further in [198] by providing the ability to pose a keyword query over a *table*. Meanwhile, in [33], queries are broken up in a federated manner, and executed over distinct, heterogeneous datasets in their native format, allowing for easy alteration of the queries and substitution of the underlying datasets being queried.

4.3 Data handling

While the "handling" that typically needs to occur for dataset search at the moment is collection and indexing of metadata, there is research in additional data handling that can improve the effectiveness of search.

Quality and entity resolution. There are several efforts dealing with metadata quality [139,175]. One solution proposed to tackle the metadata quality problem includes cross-validating metadata by merging feeds from identified entities [82]. Using self-categorized information [110] as facets is another. Attempts to better represent the underlying data [21] do have an affect on search. This includes better links with

[6] https://www.thomsonreuters.com/en.html.

[7] https://opencorporates.com/.

[8] https://www.researchably.com/.

other data [56]. Other approaches, such as [8], investigate how to detect dataset coverage and bias that could affect any algorithms that use the dataset as input.

In the context of constructive dataset search, the Mannheim Search Join Engine [114,115] and WikiTables [20] use a table similarity approach for table extension but also look at the unconstrained task. In both cases, a similarity ranking between the input and augmentation tables is used to decide which columns should be added. Interestingly, the Mannheim system also consolidates columns from different potential augmentation tables before performing the table extension.

Summarization and annotation. To help both search and user understanding, summaries and annotations are additional metadata that can be generated about the underlying dataset [105]. For instance, [136] deal with the problem that the underlying dataset cannot be exposed, but good summaries may help the user undertake the task of data access. Meanwhile, [124] use annotations to help support searching over data types and entities within a dataset, while [93] provide better labeling for numerical data in tables.

4.4 Results presentation

Ranking Datasets. Intuitively datasets require different ranking approaches due to their unique properties, which is also indicated in initial exploratory studies with users (e.g., [99,106]). Noy et al. [141] describe that links between datasets are still rare, which makes traditional web-based ranking difficult. There is some work that looks at ranking datasets. For instance, after performing a keyword query over tables, a ranking on the returned tables is attempted [198]. In a more advanced method, Van Gysel et al. [176] use an unsupervised learning approach to identify topics of a database that can then be used in ranking. Finally, [118] rank datasets containing continuous information.

Interactions. Interactive query interfaces allow ad-hoc data analysis and exploration. Facilitating users exploration changes the fundamental requirements of the supporting infrastructure with respect to processing and workload [91]. Choosing a dataset greatly depends on the information provided alongside it. A number of studies indicate that standard metadata does not provide sufficient information for dataset reuse [105,140]. Recent studies have discussed textual [105,172] or visual [183] surrogates of datasets that aim to help people identify relevant documents and increase accuracy and/or satisfaction with their relevance judgments.

There has been additional research in how to help users interact with datasets for better understanding. For instance, there is the *many-answer problem*: users struggle to specify exact queries without knowing the data and their need to understand what is available in the whole result set to formulate and refine queries [126]. Currently dataset search is mainly performed over metadata, so the users understanding of what the dataset contains before download is limited by the quality, comprehensiveness and nature of metadata. A number of frameworks or SERP designs have been proposed as research prototypes for data search and exploration, such as TableLens [152], DataLens [126], the relation browser [130] for sensemaking with statistical data, or summarization approaches of aggregate query answers in databases [181]. Navigational structures can support the cognitive representation of information [157], and we see a large space to explore interfaces that allow more complex interaction with datasets such as sophisticated querying [89] (e.g., taking a dataset as input and searching for similar ones) or being able to follow links between entities in datasets.

Interaction characteristics for dataset search have been subject to several recent human data interaction studies. Moving beyond search as a technological problem, Gregory et al. [68] show that there are also social considerations that impact a user when searching. In a comparison between document retrieval and dataset retrieval, Kern and Mathiak [99] show that users are more reliant on metadata when performing dataset search. While looking at dataset users of varying abilities [26] show that the amount of tool support can impact a user's ability to effectively discover and use a dataset. Finally, in a framework for Human Interaction with Structured data [106] discuss three major aspects that matter to data practitioners when selecting a dataset to work with: *relevance*, *usability* and *quality*. Users judge the relevance of datasets for a specific task based on the dataset's scope (e.g., geographical and temporal scope) [95,138], basic statistics about the dataset such as counts and value ranges, and information about granularity of information in the data [105]. The documentation of variables and the context from which the dataset comes from also play a key role. Data quality is intertwined with a user's assessment of "fitness for use" and depends on various factors (dimensions or characteristics) such as accuracy, timeliness, completeness, relevancy, objectivity, believability, understandability, consistency, conciseness, availability and verifiability [105]. Provenance is a prevalent attribute to judge a datasets quality as it gives an indication of the authoritativeness, trustworthiness, context and original purpose of a dataset, e.g., [84,105,135]. In order to judge a dataset's usability for a given task, the following attributes have been identified as important: format, size, documentation, language (e.g., used in headers or for string values), comparability (e.g., identifiers, units of measurement), references to connected sources, and access (e.g., license, API) [84,105,184]. These are attributes independent of a dataset's content or topical relevance which can influence whether a user is actually able to engage with the dataset.

Table 2 Mapping of Dataset search open problems to possible solution areas. We identify relevant works from other search sub-communities that could be used as inspiration for solving current dataset search problems

	Query languages Beyond keyword	Query handling Differentiated access	Query handling Extra knowledge	Results presentation Interactivity
Entity-based	[6,43,73]	[45,102]	[122,149,153]	[142,192,193]
IR		[10,134]		[78,98,104,165]
Databases	[44,59]	[18,55,88,96,150]	[1,21,36,53,62,86,123,126,133,137,145,169,191,200]	[32]
Hidden/deep web	[30,113,164]	[154]	[52,107,119]	[101]
Tabular	[63,196]		[114,115,190]	

5 Open problems

In this survey, we have organized the literature into a framework that reflects the high-level steps necessary to implement a dataset search system. We have considered current research explicitly targeting dataset search challenges. In this section, we discuss several cross-cutting themes that need to be explored in greater detail to advance dataset search.

Issues of discoverability of open data were recognized by the European Commission which oversees the process of data publishing within Europe. In 2011 they defined six barriers that challenge the reuse and true openness of data, which also apply to dataset search [58]:

- a lack of information that certain data actually exists and is available;
- a lack of clarity of which public authority holds the data;
- a lack of clarity about the terms of reuse;
- data which is made available only in formats that are difficult or expensive to use;
- complicated licensing procedures or prohibitive fees;
- exclusive reuse agreements with one commercial actor or reuse restricted to a government-owned company.

In addition to these challenges, we identify several additional problems that need attention. In order to tackle these problems, we look at similar solutions used by other search sub-areas, as described in Sect. 2.3. We map the problems we have identified in Dataset Search to solutions utilized in other search techniques that could help make headway in each problem area, as summarized in Table 2.

5.1 Query languages: moving beyond keywords

Existing dataset search systems, whether it is Google's Dataset Search or vertical engines such as those used within data repositories, use query languages and concepts from information retrieval. Information needs are expressed via keyword queries, or, in the case of faceted search, via a series of filters modeled after metadata attributes such as domain, format or publisher. Studies in tabular search point to the need for alternative interfaces, which allow users to start their search journey with a table and then add to it as they explore the results. In addition to having different ways to capture information needs, it would also be beneficial to provide query languages that are able to combine information adaptively across multiple tables. This would be especially useful for tasks such as specifying data frames or generating comprehensive data-driven reports [69].

This connects dataset search to the area of text databases [90] and the deep web. However, much of that work has looked at verticals instead of search across datasets coming from multiple domains. The problem here is to be able to identify relevant tables for the input query, join them appropriately, and do subsequent query processing.

Existing research has primarily focused on structured queries (SQL, SPARQL) over the metadata of the datasets, without considering the actual content of the dataset. There is thus a need for richer query languages that are able to go beyond the metadata of datasets and are supported by indexing systems. Our understanding of the level of expressiveness of these languages is still fairly limited. The W3C CSV on the Web working group [170] has made a proposal for specifying the semantics of columns and values in tables, but the approach requires mappings between a column and the intended semantic meaning, which are typically specified manually. Recently, the Source Retrieval Query Language (SRQL) has been proposed that facilitates declarative search for data sources based the relations of the underlying data [31].

5.1.1 Entity-centric search building blocks

Entity-centric search naturally fits within the needs of dataset search. Datasets themselves are often built of entities, and as such need the ability to specify an entity as a query, a set of entities, or a type of entity. Moreover, the notion of similarity [198] among entities should be expanded so that the entities themselves are not the focus of the match, but the number of similarities within the dataset.

5.1.2 Database building blocks

Querying datasets will likely require new adaptations to query languages and methods. In addition to the exploration of a structured query language that can operate over datasets natively, other mechanisms to define queries should be explored. For instance, the overlap of programming languages and database query languages in which programming language concepts are used to define queries over databases with different levels of capabilities [44] or over MapReduce frameworks [59], could be one such rich area to explore.

5.1.3 Tabular search building blocks

Tabular search provides an interesting view on the potential query language requirements for dataset search, where instead of keywords, the input is a table itself. This also makes novel user interfaces possible, for example, to provide assistance during the creation of spreadsheets [196].

5.2 Query handling: differentiated access

Most dataset search systems today either work within the confines of a single organization or on publicly available datasets that publish metadata according to a specified schema. However, there is demand to be able to pool information stemming from different organizations, for example, to be able to build cohorts for health studies from across clinical studies [46,136]. Providing such *differentiated access* is critical for the emerging notion of *data trusts*,[9] which provide the legal, technical and operational structures to share data between organizations.

We must facilitate an organizational as well as technical space to share data between both public and private entities. Thus, there are critical issues to be solved with respect querying over datasets with differing legal, privacy and even pricing properties. Without being able to search over these hidden datasets, access to a majority of data will be prevented. Here, aspects of using the provenance of data could be leveraged at query time [187]. We note that this is not just an issue for private data. Public data also have different properties (e.g., licenses) that users want to effectively integrate in their searches.

At an implementation level, further investigation into integrating security techniques in the query handling process is necessary, for example, searching over encrypted datasets [12,109] or using digests to minimize disclosure while still enabling search [136]. All of this must be done while also considering that the demands of reuse may change the underlying requirements and bottlenecks of query processing [61].

[9] https://theodi.org/article/what-is-a-data-trust/.

5.2.1 Information retrieval building blocks

In the context of dataset retrieval, the basic concepts supporting general web search are not sufficient, which indicates a need for a more targeted approach for dataset retrieval, treating it as a unique vertical [28,65].

5.2.2 Database building blocks

The relational algebra that underpins our processing within a database [42] has no equivalent yet in dataset search. Recently, Apache released information about the query processing system used for many of the Apache products including Hive and Storm, and Begoli et al. [18] investigated how the relational algebra can be applied to data contained within the various data processing frameworks in the Apache suite. Alternatively, other recent work in query processing attempts to handle non-relational operators via adaptive query processing [96].

Techniques such as those found in [150] suggest using a hybrid version of approximate query processing over samples and precomputation. Solutions such as ORCHESTRA [88] that were built to manage shared, structured data with changing schemas, cleaning, and queries that utilize provenance and annotation information (discussed in more detail below) need to be adapted to the dataset search problem. Other work from the probabilistic database area could also be of assistance. For instance [55] calculates the top-k results for queries over a probabilistic database by taking into account the lineage of each tuple. This usage of provenance to influence the overall ranking of the end result could inform dataset ranking.

Focusing on constructive dataset search, in which datasets are generated on-the-fly based on a user's needs and query, the work in data integration is particularly important. Querying sources in an integrated fashion [75,108] becomes a foundational component of constructive dataset search.

5.3 Data handling: extra knowledge

In order to support the differentiated access and advanced exploratory interfaces articulated above, dataset search engines will need to become more advanced in their ingestion, indexing and cataloging procedures. This problem divides into two areas: incorporation of external knowledge in the data handling process and better management and usage of dataset-intrinsic information. As described in [141], links between datasets are still rare, making identification and usage of extra knowledge difficult.

Incorporating external knowledge, whether through the use of domain ontologies, external quality indicators or even unstructured information (i.e., papers) that describe the

datasets, is a critical problem. A concrete example of this problem: many datasets are described through code books that are written in natural language. These datasets are nearly useless without integration of external information about the codebooks themselves.

Utilizing dataset-intrinsic information, is necessary to more fully capture the richness of each dataset, and allow users to express a richer set of criteria during search. Within this space, there are open problems related to data *pre-processing*. How to do quality assessment on the fly? What kinds of indexes around quality need to be created? Moving beyond quality, in general, the automatic creation and maintenance of metadata that describes datasets is difficult. Users rely up on metadata to chose appropriate datasets. Open problems for metadata include:

1. identifying the metadata that is of highest value to users w.r.t. datasets;
2. tools to automatically create and maintain that metadata;
3. automatic annotation of dataset with metadata—linking them automatically to global ontologies.

In addition to pre-processing, current dataset search systems primarily rely on information retrieval architectures (e.g., indexing into ElasticSearch) to index and perform queries. Here, lessons learned from database architectures could be applied. This is particularly the case as we have seen the importance of lessons learned from relational query engines being applied in the case of distributed data environments [7]. Thus, we think an important open problem is what the most effective *architectures* are for dataset search systems.

5.3.1 Entity-centric search building blocks

One can apply the Linked Data paradigm to solve dataset search by converting datasets to RDF and following the full cycle, as described in [110]. However, for data publishers, it is often still very expensive to execute this full cycle. Furthermore, there is debate on whether certain datasets should have an RDF representation at all, as their original formats are perhaps more suited to the tools that are required for them (e.g., geospatial datasets). A middle-ground solution is to consider datasets as resources and encode only their description in RDF, for example, using the Data Catalog Vocabulary (a W3C recommendation) [127]. Then, the Linked Data cycle can be applied to these descriptions, ultimately enabling the querying of datasets. The main challenge is the generation and maintenance of these descriptions, with some works tackling the problem of extracting specific properties from specific formats, like [138] for extracting spatio-temporal properties, and, e.g., [94] for identifying the numerical properties in CSV tables.

5.3.2 Database building blocks

As noted in [11], users do not have the "attention" to introspect deeply into large and changing datasets. Instead, we can draw upon several areas of research from the database community, including data profiling and data quality.

Naumann's recent survey [137] provides a good overview of data profiling activities based on how data-users approach the task, and what resources are available for it. Of particular note for dataset search is the work on outlier detection [53,126] as a way to provide indications to an end-user about the scope, spread and variety of a dataset during search. In particular, we note the techniques found in [200] are interesting for dataset search in that they split a large dataset into many smaller datasets and create an approximate representation of it for more accurate sampling of these subpieces. Finally, [62] establishes a tool that can comb through semi-structured log datasets to pull information into multi-layered structured datasets. All of these techniques may aid users in exploring and making sense of dataset. Given that a dataset is by definition a collection of pieces, imputation of missing pieces needs great scrutiny. As discussed in Sect. 4, imputation efforts are underway [1,21,123,169] but draw heavily from web techniques. The imputation methods from the data management community should be considered.

The work on profiling contains expressions of data cleanliness and coverage, completeness and consistency. These properties are classic data quality metrics, and help the user form a picture of whether the data is fit for use. Automatic understanding of data quality in order to either populate metadata or answer metadata queries in a lazy manner will require techniques that can automatically determine complex datatypes such as [191]. Currently, though, the research in each of these areas has been focused on its relationship to describing or working within a specific artifact, not as a component for a search. To do this, the structures and content for each area need to be computable in a timely manner and presented in a way that can be taken advantage of by a search system. For instance, data quality is a traditionally resource expensive task that is often domain-specific. Generic, albeit possibly less accurate methods must be developed to compute data quality estimations that can be accessed and used during search [36,133].

In order to facilitate understanding of the contents of a dataset, summarization can be used, as done in [145] over probabilistic databases. Provenance, another tools that could help users understand a dataset, has an unsolved problem of moving across granularity levels. A tuple within a dataset may have provenance associated with it, as may the table,

and the entire dataset itself. The challenge is in understanding how the aggregation of tuple-provenance would affect the search results compared to dataset-provenance. Finally, using annotations to improve the data [86] will be needed. Interesting extensions could include using user feedback to facilitate ranking of datasets based on the searcher's criteria, or utilizing the context under which the annotations were created to change how annotations impact ranking.

5.3.3 Hidden/deep web building blocks

An inherent challenge in dataset search over the web is to be able to identify particular resources as datasets of interest (and ignore, for example, natural language documents). This challenge will be also present in any forthcoming approach in searching for datasets on the deep web. Moreover, any such approach will build on some combination of the two main directions for surfacing deep web data. Building vertical engines for the hidden web has the difficulties of predefining all interesting domains, identifying relevant forms in front of datasets on the web and investigating automatic (or semi-automatic) approaches to create mappings; a task which seems extremely hard on a web scale. Hence, learning/computing web form inputs might be the option of choice. Nevertheless, in cases where there are complex domains that involve many attributes and involved inputs, e.g., airline reservations, when the datasets change frequently, e.g., financial data, or when forms use the http POST method [128] virtual integration remains an attractive direction.

5.3.4 Tabular search building blocks

The majority of work in tabular search addresses web tables, not uploaded datasets. These tables have the benefit of generally being better described and often general-knowledge related, e.g., column names are human readable and not codes, or the tables are embedded in larger documents (e.g., HTML tables). In addition, a majority of work treats what are termed 'entity-centric tables', which are tables in which each row represents a single entity. Datasets can be much more general, for example, containing multiple tables in one file.

5.4 Result presentation: interactivity

As previously discussed, existing data search systems follow similar approaches to search showing a ranked list of search results with some additional faceted searching in place. At a tactical level, ranking approaches specifically tailored to dataset search should be developed. Importantly, this should take into account the kinds of rich indexes suggested in the prior section. Here, the challenges are that typical approaches to improving ranking in information retrieval such as learning to rank are difficult given that many data search engines do not have the kind of level of user traffic needed for learning to rank algorithms [176]. In addition, the integration of dataset search and entity search is an important open problem. For example, when searching for a chemical we could also display associated data, but we currently know little about what data that should be. Beyond standard search paradigms, supporting conversational search over data and embedding search into the actual data usage process deserves significant attention, particularly since dataset search is often needed in the context of a variety of tasks [167].

5.4.1 Information retrieval building blocks

As pointed out by Cafarella et al. [28] structured data on the web is similar to the scenario of ranking of millions of individual databases. Tables available online contain a mixture of structural and related content elements which cannot easily be mapped to unstructured text scenarios applied in general web search. Tables lack the incoming hyperlink anchor text and are two-dimensional—they cannot be efficiently queried using the standard inverted index. For those reasons PageRank-based algorithms known from general web search are not applicable to the same extent to the dataset/table search, particularly as tables of widely-varying quality can be found on a single web page.

Search for datasets is often complex and shows characteristics of exploratory search tasks, involving multiple queries, iterations and refinement of the original information need, as well as complex cognitive processing [106]. There are many possible reasons that users have diverse interaction styles, from context and domain specificity [68] to uncertainty in the search workflow itself [26]. It is important to note that users have different interaction styles with respect to "getting the data". These interactions range from question answering to "data return" to exploration [68,106]. From an interaction perspective, dataset search is not as advanced as web or document search. Contextual or personalized results, which are common on the web [182] are practically non-existent for dataset search. Additionally, as mentioned, dataset search relies on limited metadata instead of looking at the dataset itself which limits interaction. While many classifications for information seeking tasks exist [22], there is no widely used classification of data-centric information seeking tasks yet that could be used to model interaction flows in dataset search.

5.4.2 Database building blocks

Provenance [27,67,81,187] is likely to be a key element in assisting the user in choosing a dataset of interest. Until now, provenance has been used to facilitate trust in an artifact

[47,48] or automatically estimate quality [85]. New methods must be developed to facilitate translation of this large graph into a format that a user who is evaluating whether or not to use a dataset can interpret and utilize [34]. The logic and possible new operators behind dataset search will open up new areas for determining why and why not to consider provenance of the dataset query results themselves [35,81,112].

The presentation of data models has been a topic in database literature [89] as well as exploration strategies of result spaces beyond the 10 blue links paradigm. For instance, the use of sideways and downwards exploration of web table queries by [39]. Challenges and directions for search results presentation and data exploration as part of the search process are discussed on a mostly speculative basis in the literature and include representing different types of results in a manner that express the structure of the underlying dataset (tables, networks, spatial presentations, etc) [89].

An overview of search results can enhance orientation and understanding of the information provided [157], which allows to get an awareness of the dataset result space as a whole. Making a large set of possible results more informative to the user has been explored for databases [181]. At the same time being able to investigate the dataset on a column, row and cell level to match both process and content oriented requirements on the search result can be necessary [151,170].

Within the scope of constructive dataset search, the work of [186] is essential to appropriately annotate and cite the results of queries.

In the next section, we discuss one foundation that is crucial for addressing these open problems, benchmarks.

6 The road forward: benchmarks

One of the most widely recognized problems of dataset search is the lack of benchmarks. For instance, the BioCADDIE project, which attempts to index for discovery scientific datasets, has a pilot project to recommend appropriate datasets to users based on similar topic, size, usage, user background and context [92]. In order to do this, the pilot participants are creating a topic model across scientific articles, and using user query patterns to identify similar users. While this is an interesting start, and acknowledges that there are a myriad of overlapping concerns that impact dataset search, from content to the user's ability, there is no way yet to measure whether the solution works. For this, a clear benchmark is needed. In this section we will outline the state of the art with respect to the evaluation of different parts of the dataset search pipeline, which were discussed earlier in this work.

Step one is identifying the set of metrics that are appropriate to dataset search. Do they mimic the online and offline metrics of information retrieval? At first blush, session abandonment rate, session success rate and zero result rate from information retrieval online metrics appear relevant, while click-through rate may need some adjustment for the context of datasets. Meanwhile, most of the offline metrics, from the set of precision-based metrics, to recall, fall-out, discounted cumulative gain, etc. are obviously still necessary.

However, there are dataset-specific metrics that may need to be considered. For instance, "completeness" could be an interesting new metric to consider. Many tasks involving datasets require the stitching of several datasets to create a whole that is fit for purpose. Is the right set, that creates a "complete" offering returned? How do we measure that the appropriate set of datasets for a given purpose were returned. For instance, in the context of information retrieval on an Open Data Platform, [95] found that some user queries require multiple datasets which are equally relevant in opposition to a ranked result list of resources with single resource per rank. The question of how such result list should be returned to the user remains open, and creates an interesting case within benchmark creation. To facilitate interactive dataset retrieval studies we would need to have a clearer understanding of selection criteria for datasets, a taxonomy of data-centric tasks and annotated corpora of information tasks for datasets, queries and connected *relevant* datasets as search results.

The availability of benchmarks upon which solutions across the query processing pipeline for dataset search can be tested is essential. Any benchmark created for dataset search needs to, explicitly or implicitly, highlight the relationships that exist between the user, the task at hand and the properties of the dataset or it's metadata. Unlike classic web retrieval, there are added dimensions for dataset search. It is not enough for a user to find the information appropriate based on its content; for dataset search, the user and the specific task requirements must be satisfied. The result list presented to the user must be understandable and explorable, due to the added complexity of interpreting and using data.

Several benchmarks have already been created that cover tasks related to dataset search. These benchmarks include: managing RDF datasets [146]; information retrieval over Wikipedia tables [198]; assignment of semantic labels to web tables [158]. Further efforts in this area are needed in order to truly understand and make progress on the underlying technology.

Moreover, the availability of benchmarks will enable performance evaluations across search architectures, enabling a better ability for tool users to choose an appropriate solution for their specific needs. Ultimately, through benchmarks, and performance evaluations, we should be able to design data search systems that assist a user, for example, who needs to search for a dataset to do a particular classification task, and let that user clearly understand which methods will provide

the best results on the returned dataset or which risks might be associated with using a particular dataset.

7 Conclusions

The topic of data-driven research will only grow; we are at the start of a journey in which datasets are used for analysis, decision making and resource optimization, am. Our current needs for dataset search require us to give due attention to this problem. The current state of the art is focused on tuple, document or webpage. Datasets are an interesting entity to themselves with some properties shared with documents, tuples and webpages, and some unique to datasets.

In this work, we highlight that dataset search can be achieved through two different mechanisms: (1) issue query, return dataset; (2) issue query, build dataset. However, dataset search itself is in its infancy. Techniques from many other fields, including databases, information retrieval, and semantic web search, can be applied toward the problem of dataset search. The creation of an initial service, Google Dataset Search, that allows for automatic indexing of datasets, and Google-style search over that indexed information marks this problem as important. Moreover, it highlights the research that still needs to be performed within the dataset retrieval domain, including: formal query language(s), dealing with social and organizational restrictions when processing a query, providing additional information to support query processing, facilitating user exploration and interaction with a result set made up of datasets. This is an exciting time with respect to dataset search, in which there is a high need for datasets of all sorts, combined with burgeoning tools for dataset search, like Google Dataset Search, that provide the necessary infrastructure. However, further research is needed to fully understand and support dataset search.

Acknowledgements The following projects supported this work: TheyBuyForYou (EU H2020: 780247); Data Stories (EPRSC: EP/P025676/1); QROWD (EU:H2020 732194); A Multidisciplinary Study of Predictive Artificial Intelligence Technologies in the Criminal Justice System (Alan Turing Institute RCP009768).

References

1. Ahmadov, A., Thiele, M., Eberius, J., Lehner, W., Wrembel, R.: Towards a hybrid imputation approach using web tables. In: 2015 IEEE/ACM 2nd International Symposium on Big Data Computing (BDC), pp. 21–30. IEEE (2015). https://doi.org/10.1109/BDC.2015.38
2. Ai, Q., Dumais, S.T., Craswell, N., Liebling, D.: Characterizing email search using large-scale behavioral logs and surveys. In: Proceedings of the 26th International Conference on World Wide Web, WWW '17, pp. 1511–1520. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2017). https://doi.org/10.1145/3038912.3052615
3. Alahakoon, D., Yu, X.: Smart electricity meter data intelligence for future energy systems: a survey. IEEE Trans. Ind. Inform. **12**(1), 425–436 (2016). https://doi.org/10.1109/TII.2015.2414355
4. Alexe, B., ten Cate, B., Kolaitis, P.G., Tan, W.C.: Designing and refining schema mappings via data examples. In: Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 2011), pp. 133–144. Athens, Greece (2011)
5. Altman, M., Castro, E., Crosas, M., Durbin, P., Garnett, A., Whitney, J.: Open journal systems and dataverse integration—helping journals to upgrade data publication for reusable research. Code4Lib J. **30** (2015)
6. Angles, R., Arenas, M., Barceló, P., Hogan, A., Reutter, J., Vrgoč, D.: Foundations of modern query languages for graph databases. ACM Comput. Surv. **50**(5), 68:1–68:40 (2017). https://doi.org/10.1145/3104031
7. Armbrust, M., Xin, R.S., Lian, C., Huai, Y., Liu, D., Bradley, J.K., Meng, X., Kaftan, T., Franklin, M.J., Ghodsi, A., et al.: Spark sql: relational data processing in spark. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1383–1394. ACM (2015). https://doi.org/10.1145/2723372.2742797
8. Asudeh, A., Jin, Z., Jagadish, H.V.: Assessing and remedying coverage for a given dataset. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE), pp. 554–565 (2019). https://doi.org/10.1109/ICDE.2019.00056
9. Auer, S., Bühmann, L., Dirschl, C., Erling, O., Hausenblas, M., Isele, R., Lehmann, J., Martin, M., Mendes, P.N., Van Nuffelen, B., Stadler, C., Tramp, S., Williams, H.: Managing the life-cycle of linked data with the LOD2 stack. In: International semantic Web conference, pp. 1–16. Springer (2012). https://doi.org/10.1007/978-3-642-35173-0_1
10. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern information retrieval—the concepts and technology behind search, 2nd edn. Pearson Education Ltd., Harlow (2011). http://www.mir2ed.org/
11. Bailis, P., Gan, E., Rong, K., Suri, S.: Prioritizing attention in fast data: principles and promise. In: Conference on Innovative Dataset Research (CIDR) (2017)
12. Bakshi, S., Chavan, S., Kumar, A., Hargaonkar, S.: Query processing on encoded data using bitmap. J. Data Min. Manag. **3** (2018)
13. Balazinska, M., Howe, B., Koutris, P., Suciu, D., Upadhyaya, P.: A Discussion on Pricing Relational Data, pp. 167–173. Springer, Berlin (2013). https://doi.org/10.1007/978-3-642-41660-6_7
14. Balog, K.: Entity-Oriented Search. Springer, Berlin (2018)
15. Balog, K., Meij, E., de Rijke, M.: Entity search: building bridges between two worlds. In: Proceedings of the 3rd International Semantic Search Workshop, SEMSEARCH '10, pp. 9:1–9:5. ACM, New York, NY, USA (2010). https://doi.org/10.1145/1863879.1863888
16. Balog, K., Serdyukov, P., de Vries, A.P.: Overview of the TREC 2010 entity track. In: TREC (2010)
17. Batty, M.: Big data and the city. Built Environ. **42**, 321–337 (2016). https://doi.org/10.2148/benv.42.3.321
18. Begoli, E., Camacho-Rodríguez, J., Hyde, J., Mior, M.J., Lemire, D.: Apache calcite: a foundational framework for optimized query processing over heterogeneous data sources. In: Proceedings of

the 2018 International Conference on Management of Data, SIGMOD '18, pp. 221–230. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3190662
19. Bertino, E., Ooi, B.C., Sacks-Davis, R., Tan, K.L., Zobel, J., Shidlovsky, B., Andronico, D.: Indexing Techniques for Advanced Database Systems. Springer, Berlin (2012)
20. Bhagavatula, C.S., Noraset, T., Downey, D.: Methods for exploring and mining tables on wikipedia. In: Proceedings of the ACM SIGKDD Workshop on Interactive Data Exploration and Analytics, pp. 18–26. ACM (2013). https://doi.org/10.1145/2501511.2501516
21. Bischof, S., Harth, A., Kämpgen, B., Polleres, A., Schneider, P.: Enriching integrated statistical open city data by combining equational knowledge and missing value imputation. J. Web Semant. **48**, 22–47 (2018). https://doi.org/10.1016/j.websem.2017.09.003
22. Blandford, A., Attfield, S.: Interacting with information. Synth. Lect. Hum. Centered Inform. **3**(1), 1–99 (2010)
23. Bordes, A., Gabrilovich, E.: Constructing and mining web-scale knowledge graphs: Kdd 2014 tutorial. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, pp. 1967–1967. ACM, New York, NY, USA (2014). https://doi.org/10.1145/2623330.2630803
24. Borgman, C.L.: The conundrum of sharing research data. J. Am. Soc. Inf. Sci. Technol. **63**(6), 1059–1078 (2012). https://doi.org/10.1002/asi.22634
25. Borgman, C.L.: Big Data, Little Data. Scholarship in the Networked World. The MIT Press, Cambridge (2015)
26. Boukhelifa, N., Perrin, M.E., Huron, S., Eagan, J.: How data workers cope with uncertainty: a task characterisation study. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17, pp. 3645–3656. ACM, New York, NY, USA (2017). https://doi.org/10.1145/3025453.3025738
27. Buneman, P., Chapman, A., Cheney, J.: Provenance management in curated databases. In: Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data, SIGMOD '06, pp. 539–550. ACM, New York, NY, USA (2006). https://doi.org/10.1145/1142473.1142534
28. Cafarella, M.J., Halevy, A., Wang, D.Z., Wu, E., Zhang, Y.: Webtables: exploring the power of tables on the web. Proc. VLDB Endow. **1**(1), 538–549 (2008). https://doi.org/10.14778/1453856.1453916
29. Cafarella, M.J., Halevy, A.Y., Lee, H., Madhavan, J., Yu, C., Wang, D.Z., Wu, E.: Ten years of webtables. PVLDB **11**(12), 2140–2149 (2018). https://doi.org/10.14778/3229863.3240492
30. Calì, A., Martinenghi, D.: Querying the deep web. In: Proceedings of the 13th International Conference on Extending Database Technology, EDBT '10, pp. 724–727. ACM, New York, NY, USA (2010). https://doi.org/10.1145/1739041.1739138
31. Castro Fernandez, R., Abedjan, Z., Koko, F., Yuan, G., Madden, S., Stonebraker, M.: Aurum: a data discovery system. In: 2018 IEEE 34th International Conference on Data Engineering (ICDE), pp. 1001–1012 (2018). https://doi.org/10.1109/ICDE.2018.00094
32. Catarci, T.: What happened when database researchers met usability. Inf. Syst. **25**(3), 177–212 (2000). https://doi.org/10.1016/S0306-4379(00)00015-6
33. Chamanara, J., König-Ries, B., Jagadish, H.V.: Quis: in-situ heterogeneous data source querying. Proc. VLDB Endow. **10**(12), 1877–1880 (2017). https://doi.org/10.14778/3137765.3137798
34. Chapman, A., Blaustein, B.T., Seligman, L., Allen, M.D.: Plus: a provenance manager for integrated information. In: 2011 IEEE International Conference on Information Reuse Integration, pp. 269–275 (2011). https://doi.org/10.1109/IRI.2011.6009558
35. Chapman, A., Jagadish, H.V.: Why not? In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, SIGMOD '09, pp. 523–534. ACM, New York, NY, USA (2009). https://doi.org/10.1145/1559845.1559901
36. Chapman, A.P., Rosenthal, A., Seligman, L.: The challenge of quick and dirty information quality. J. Data Inf. Qual. **7**(1–2), 1:1–1:4 (2016). https://doi.org/10.1145/2834123
37. Chaudhuri, S.: An overview of query optimization in relational systems. In: Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 34–43. ACM (1998)
38. Chen, J., DeWitt, D.J., Tian, F., Wang, Y.: Niagaracq: a scalable continuous query system for internet databases. ACM SIGMOD Rec. **29**, 379–390 (2000)
39. Chirigati, F., Liu, J., Korn, F., Wu, Y.W., Yu, C., Zhang, H.: Knowledge exploration using tables on the web. Proc. VLDB Endow. **10**(3), 193–204 (2016). https://doi.org/10.14778/3021924.3021935
40. Christophides, V., Efthymiou, V.: Entity Resolution in the Web of Data. Morgan and Claypool, San Rafael (2015)
41. CKAN (2018). https://ckan.org/
42. Codd, E.F.: Relational Completeness of Data Base Sublanguages. Citeseer (1972)
43. Corby, O., Faron-Zucker, C., Gandon, F.: Ldscript: a linked data script language. In: d'Amato, C., Fernandez, M., Tamma, V., Lecue, F., Cudré-Mauroux, P., Sequeda, J., Lange, C., Heflin, J. (eds.) The Semantic Web—ISWC 2017, pp. 208–224. Springer, Cham (2017)
44. Costa Seco, J., Ferreira, P., Lourenço, H.: Capability-based localization of distributed and heterogeneous queries. J. Funct. Program. **27**, e26 (2017). https://doi.org/10.1017/S095679681700017X
45. Costabello, L., Villata, S., Rodriguez Rocha, O., Gandon, F.: Access control for http operations on linked data. In: Cimiano, P., Corcho, O., Presutti, V., Hollink, L., Rudolph, S. (eds.) The Semantic Web: Semantics and Big Data, pp. 185–199. Springer, Berlin (2013)
46. Cui, L., Zeng, N., Kim, M., Mueller, R., Hankosky, E.R., Redline, S., Zhang, G.Q.: X-search: an open access interface for cross-cohort exploration of the national sleep research resource. BMC Med. Inform. Decis. Mak. **18**(1), 99 (2018). https://doi.org/10.1186/s12911-018-0682-y
47. Curcin, V., Fairweather, E., Danger, R., Corrigan, D.: Templates as a method for implementing data provenance in decision support systems. J. Biomed. Inform. **65**, 1–21 (2017). https://doi.org/10.1016/j.jbi.2016.10.022
48. Dai, C., Lin, D., Bertino, E., Kantarcioglu, M.: An approach to evaluate data trustworthiness based on data provenance. In: Jonker, W., Petković, M. (eds.) Secure Data Management, pp. 82–98. Springer, Berlin (2008)
49. Dalvi, B.B., Cohen, W.W., Callan, J.: Websets: extracting sets of entities from the web using unsupervised information extraction. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, WSDM '12, pp. 243–252. ACM, New York, NY, USA (2012). https://doi.org/10.1145/2124295.2124327
50. d'Aquin, M., Ding, L., Motta, E.: Semantic Web Search Engines, pp. 659–700. Springer, Berlin (2011)
51. Das Sarma, A., Fang, L., Gupta, N., Halevy, A., Lee, H., Wu, F., Xin, R., Yu, C.: Finding related tables. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, pp. 817–828. ACM (2012). https://doi.org/10.1145/2213836.2213962
52. Deng, S.: Deep web data source selection based on subject and probability model. In: 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). IEEE (2016). https://doi.org/10.1109/imcec.2016.7867557

53. Dong, B., Wang, H.W., Monreale, A., Pedreschi, D., Giannotti, F., Guo, W.: Authenticated outlier mining for outsourced databases. IEEE Trans. Dependable Secur. Comput. (2017). https://doi.org/10.1109/TDSC.2017.2754493
54. Dong, X.L.: Challenges and innovations in building a product knowledge graph. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '18, pp. 2869–2869. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3219819.3219938
55. Dylla, M., Miliaraki, I., Theobald, M.: Top-k query processing in probabilistic databases with non-materialized views. In: 2013 IEEE 29th International Conference on Data Engineering (ICDE), pp. 122–133 (2013). https://doi.org/10.1109/ICDE.2013.6544819
56. Ellefi, M.B., Bellahsene, Z., Dietze, S., Todorov, K.: Dataset recommendation for data linking: an intensional approach. In: International Semantic Web Conference, pp. 36–51. Springer (2016)
57. Elsevier scientific repository (2018). https://datasearch.elsevier.com/
58. European Commission, D.A.: Commission's open data strategy, questions and answers. Memo/11/891 (2011)
59. Fegaras, L.: An algebra for distributed big data analytics. J. Funct. Program. **27**, e27 (2017). https://doi.org/10.1017/S0956796817000193
60. Freitas, A., Curry, E., Oliveira, J.G., O'Riain, S.: Querying heterogeneous datasets on the linked data web: challenges, approaches, and trends. IEEE Internet Comput. **16**(1), 24–33 (2012)
61. Galakatos, A., Crotty, A., Zgraggen, E., Binnig, C., Kraska, T.: Revisiting reuse for approximate query processing. Proc. VLDB Endow. **10**(10), 1142–1153 (2017). https://doi.org/10.14778/3115404.3115418
62. Gao, Y., Huang, S., Parameswaran, A.: Navigating the data lake with datamaran: automatically extracting structure from log datasets. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, pp. 943–958. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3183746
63. Gentile, A.L., Kirstein, S., Paulheim, H., Bizer, C.: Extending rapidminer with data search and integration capabilities. In: Sack, H., Rizzo, G., Steinmetz, N., Mladenić, D., Auer, S., Lange, C. (eds.) The Semantic Web, pp. 167–171. Springer, Cham (2016)
64. Gohar, M., Muzammal, M., Rahman, A.U.: SMART TSS: defining transportation system behavior using big data analytics in smart cities. Sustain. Cities Soc. **41**, 114–119 (2018). https://doi.org/10.1016/j.scs.2018.05.008
65. Gonzalez, H., Halevy, A.Y., Jensen, C.S., Langen, A., Madhavan, J., Shapley, R., Shen, W., Goldberg-Kidon, J.: Google fusion tables: web-centered data management and collaboration. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, SIGMOD '10, pp. 1061–1066. ACM, New York, NY, USA (2010). https://doi.org/10.1145/1807167.1807286
66. Google: Google dataset search (2018). https://developers.google.com/search/docs/data-types/dataset
67. Green, T.J., Karvounarakis, G., Tannen, V.: Provenance semirings. In: Proceedings of the Twenty-sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS '07, pp. 31–40. ACM, New York, NY, USA (2007). https://doi.org/10.1145/1265530.1265535
68. Gregory, K., Groth, P.T., Cousijn, H., Scharnhorst, A., Wyatt, S.: Searching data: a review of observational data retrieval practices (2017). CoRR arXiv:1707.06937
69. Groth, P.T., Scerri, A., Jr., R.D., Allen, B.P.: End-to-end learning for answering structured queries directly over text (2018). CoRR arXiv:1811.06303
70. Grubenmann, T., Bernstein, A., Moor, D., Seuken, S.: Financing the web of data with delayed-answer auctions. In: Proceedings of the 2018 World Wide Web Conference, WWW '18, pp. 1033–1042. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2018). https://doi.org/10.1145/3178876.3186002
71. Guha, R.V., Brickley, D., Macbeth, S.: Schema.org: evolution of structured data on the web. Commun. ACM **59**(2), 44–51 (2016). https://doi.org/10.1145/2844544
72. Gupta, S., Szekely, P., Knoblock, C.A., Goel, A., Taheriyan, M., Muslea, M.: Karma: a system for mapping structured sources into the semantic web. In: Simperl, E., Norton, B., Mladenic, D., Della Valle, E., Fundulaki, I., Passant, A., Troncy, R. (eds.) The Semantic Web: Satellite Events, pp. 430–434. Springer, Berlin (2015)
73. Gutierrez, C., Hurtado, C.A., Mendelzon, A.O., Pérez, J.: Foundations of semantic web databases. J. Comput. Syst. Sci. **77**(3), 520–541 (2011). https://doi.org/10.1016/j.jcss.2010.04.009
74. Halevy, A., Korn, F., Noy, N.F., Olston, C., Polyzotis, N., Roy, S., Whang, S.E.: Goods: organizing google's datasets. In: Proceedings of the 2016 International Conference on Management of Data, pp. 795–806. ACM (2016)
75. Halevy, A.Y.: Answering queries using views: a survey. VLDB J. **10**(4), 270–294 (2001)
76. Hartig, O., Bizer, C., Freytag, J.C.: Executing SPARQL queries over the web of linked data. In: International Semantic Web Conference, pp. 293–309. Springer (2009)
77. He, B., Patel, M., Zhang, Z., Chang, K.C.C.: Accessing the deep web. Commun. ACM **50**(5), 94–101 (2007)
78. Hearst, M.: Search User Interfaces. Cambridge University Press, Cambridge (2009)
79. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Morgan and Claypool, San Rafael (2011)
80. Hendler, J., Holm, J., Musialek, C., Thomas, G.: Us government linked open data: Semantic.data.gov. IEEE Intell. Syst. **27**(3), 25–31 (2012). https://doi.org/10.1109/MIS.2012.27
81. Herschel, M., Diestelkämper, R., Ben Lahmar, H.: A survey on provenance: what for? what form? what from? VLDB J. **26**(6), 881–906 (2017). https://doi.org/10.1007/s00778-017-0486-1
82. Heyvaert, P., Colpaert, P., Verborgh, R., Mannens, E., Van de Walle, R.: Merging and enriching DCAT feeds to improve discoverability of datasets. In: International Semantic Web Conference, pp. 67–71. Springer (2015)
83. Hogan, A., Harth, A., Umbrich, J., Kinsella, S., Polleres, A., Decker, S.: Searching and browsing linked data with swse: the semantic web search engine. Web Semant. Sci. Serv. Agents World Wide Web **9**(4), 365–401 (2011)
84. Holland, S., Hosny, A., Newman, S., Joseph, J., Chmielinski, K.: The dataset nutrition label: a framework to drive higher data quality standards (2018). CoRR arXiv:1805.03677
85. Huynh, T., Ebden, M., Fischer, J., Roberts, S., Moreau, L.: Provenance network analytics: an approach to data analytics using data provenance. Data Min. Knowl. Discov. (2018). https://doi.org/10.1007/s10618-017-0549-3
86. Ibrahim, K., Du, X., Eltabakh, M.: Proactive annotation management in relational databases. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, SIGMOD '15, pp. 2017–2030. ACM, New York, NY, USA (2015). https://doi.org/10.1145/2723372.2749435
87. Ioannidis, Y.E.: Query optimization. ACM Comput. Surv. (CSUR) **28**(1), 121–123 (1996)
88. Ives, Z.G., Green, T.J., Karvounarakis, G., Taylor, N.E., Tannen, V., Talukdar, P.P., Jacob, M., Pereira, F.: The orchestra collaborative data sharing system. SIGMOD Rec. **37**(3), 26–32 (2008). https://doi.org/10.1145/1462571.1462577

89. Jagadish, H.V., Chapman, A., Elkiss, A., Jayapandian, M., Li, Y., Nandi, A., Yu, C.: Making database systems usable. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Beijing, China, June 12-14, 2007, pp. 13–24 (2007). https://doi.org/10.1145/1247480.1247483
90. Jain, A., Doan, A., Gravano, L.: SQL queries over unstructured text databases. In: IEEE 23rd International Conference on Data Engineering, 2007. ICDE 2007, pp. 1255–1257. IEEE (2007)
91. Jiang, L., Rahman, P., Nandi, A.: Evaluating interactive data systems: workloads, metrics, and guidelines. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, pp. 1637–1644. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3197386
92. Jiang, X., Qin, Z., Vaidya, J., Menon, A., Yu, H.: Pilot project 2.1—data recommendation using machine learning and crowdsourcing (2018)
93. Kacprzak, E., Giménez-García, J.M., Piscopo, A., Koesten, L., Ibáñez, L.D., Tennison, J., Simperl, E.: Making sense of numerical data-semantic labelling of web tables. In: European Knowledge Acquisition Workshop, pp. 163–178. Springer (2018)
94. Kacprzak, E., Giménez-Garcéa, J.M., Piscopo, A., Koesten, L., Ibáñez, L.D., Tennison, J., Simperl, E.: Making sense of numerical data–semantic labelling of web tables. In: Faron Zucker, C., Ghidini, C., Napoli, A., Toussaint, Y. (eds.) Knowledge Engineering and Knowledge Management. Lecture Notes in Computer Science, pp. 163–178. Springer, Berlin (2018)
95. Kacprzak, E., Koesten, L., Ibáñez, L.D., Blount, T., Tennison, J., Simperl, E.: Characterising dataset search—an analysis of search logs and data requests. J. Web Semant. (2018). https://doi.org/10.1016/j.websem.2018.11.003
96. Kaftan, T., Balazinska, M., Cheung, A., Gehrke, J.: Cuttlefish: a lightweight primitive for adaptive query processing (2018). CoRR arXiv:1802.09180
97. Kassen, M.: A promising phenomenon of open data: a case study of the chicago open data project. Gov. Inf. Q. **30**(4), 508–513 (2013). https://doi.org/10.1016/j.giq.2013.05.012
98. Kelly, D., Azzopardi, L.: How many results per page?: A study of serp size, search behavior and user experience. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '15, pp. 183–192. ACM, New York, NY, USA (2015). https://doi.org/10.1145/2766462.2767732
99. Kern, D., Mathiak, B.: Are there any differences in data set retrieval compared to well-known literature retrieval? In: Kapidakis, S., Mazurek, C., Werla, M. (eds.) Research and Advanced Technology for Digital Libraries, pp. 197–208. Springer, Berlin (2015)
100. Khare, R., An, Y., Song, I.Y.: Understanding deep web search interfaces: a survey. ACM SIGMOD Rec. **39**(1), 33–40 (2010)
101. Khare, R., An, Y., Song, I.Y.: Understanding deep web search interfaces: a survey. SIGMOD Rec. **39**(1), 33–40 (2010). https://doi.org/10.1145/1860702.1860708
102. Kirrane, S., Mileo, A., Decker, S.: Access control and the resource description framework: a survey. Semant. Web **8**(2), 311–352 (2016). https://doi.org/10.3233/SW-160236
103. Kitchin, R.: The real-time city? Big data and smart urbanism. GeoJournal **79**(1), 1–14 (2014). https://doi.org/10.1007/s10708-013-9516-8
104. Klouche, K., Ruotsalo, T., Micallef, L., Andolina, S., Jacucci, G.: Visual re-ranking for multi-aspect information retrieval. In: Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval, CHIIR 2017, Oslo, Norway, March 7–11, 2017, pp. 57–66 (2017). https://doi.org/10.1145/3020165.3020174
105. Koesten, L., Simperl, E., Kacprzak, E., Blount, T., Tennison, J.: Everything you always wanted to know about a dataset: studies in data summarisation (2018). CoRR arXiv:1810.12423
106. Koesten, L.M., Kacprzak, E., Tennison, J.F.A., Simperl, E.: The trials and tribulations of working with structured data: a study on information seeking behaviour. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver 2017, pp. 1277–1289 (2017). https://doi.org/10.1145/3025453.3025838
107. Kolias, V., Anagnostopoulos, I., Zeadally, S.: Structural analysis and classification of search interfaces for the deep web. Comput. J. **61**(3), 386–398 (2017). https://doi.org/10.1093/comjnl/bxx098
108. Konstantinidis, G., Ambite, J.L.: Scalable query rewriting: a graph-based approach. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 97–108. Athens, Greece (2011)
109. Kumar, A., Hussain, M.: Secure query processing over encrypted database through cryptdb. In: Sa, P.K., Bakshi, S., Hatzilygeroudis, I.K., Sahoo, M.N. (eds.) Recent Findings in Intelligent Computing Techniques, pp. 307–319. Springer, Singapore (2018)
110. Kunze, S.R., Auer, S.: Dataset retrieval. In: 2013 IEEE Seventh International Conference on Semantic Computing, pp. 1–8 (2013)
111. Kwok, C.C.T., Etzioni, O., Weld, D.S.: Scaling question answering to the web. ACM Trans. Inf. Syst. **19**(3), 242–262 (2001). https://doi.org/10.1145/502115.502117
112. Lee, S., Köhler, S., Ludäscher, B., Glavic, B.: A SQL-middleware unifying why and why-not provenance for first-order queries. In: 2017 IEEE 33rd International Conference on Data Engineering (ICDE), pp. 485–496 (2017). https://doi.org/10.1109/ICDE.2017.105
113. Lehmann, J., Furche, T., Grasso, G., Ngomo, A.C.N., Schallhart, C., Sellers, A., Unger, C., Bühmann, L., Gerber, D., Höffner, K., Liu, D., Auer, S.: DEQA: deep web extraction for question answering. In: Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., Parreira, J.X., Hendler, J., Schreiber, G., Bernstein, A., Blomqvist, E. (eds.) The Semantic Web—ISWC 2012, pp. 131–147. Springer, Berlin (2012)
114. Lehmberg, O., Bizer, C.: Stitching web tables for improving matching quality. Proc. VLDB Endow. **10**(11), 1502–1513 (2017). https://doi.org/10.14778/3137628.3137657
115. Lehmberg, O., Ritze, D., Ristoski, P., Meusel, R., Paulheim, H., Bizer, C.: The mannheim search join engine. J. Web Semant. **35**, 159–166 (2015). https://doi.org/10.1016/j.websem.2015.05.001
116. Levy, A.Y., Srivastava, D., Kirk, T.: Data model and query evaluation in global information systems. J. Intell. Inf. Syst. **5**(2), 121–143 (1995)
117. Li, F., Jagadish, H.V.: NaLIR: an interactive natural language interface for querying relational databases. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, pp. 709–712. ACM (2014)
118. Li, J., Deshpande, A.: Ranking continuous probabilistic datasets. Proc. VLDB Endow. **3**(1–2), 638–649 (2010). https://doi.org/10.14778/1920841.1920923
119. Li, X., Dong, X.L., Lyons, K., Meng, W., Srivastava, D.: Truth finding on the deep web: is the problem solved? In: Proceedings of the 39th International Conference on Very Large Data Bases, PVLDB'13, pp. 97–108. VLDB Endowment (2013). http://dl.acm.org/citation.cfm?id=2448936.2448943
120. Li, X., Liu, B., Yu, P.: Time sensitive ranking with application to publication search. In: Eighth IEEE International Conference on Data Mining, 2008. ICDM'08, pp. 893–898. IEEE (2008)
121. Li, Y., Yang, H., Jagadish, H.: NaLIX: an interactive natural language interface for querying XML. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 900–902. ACM (2005)

122. Li, Y.F., Wang, S.B., Zhou, Z.H.: Graph quality judgement: a large margin expedition. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, pp. 1725–1731. AAAI Press (2016)
123. Li, Z., Sharaf, M.A., Sitbon, L., Sadiq, S., Indulska, M., Zhou, X.: A web-based approach to data imputation. World Wide Web **17**(5), 873–897 (2014)
124. Limaye, G., Sarawagi, S., Chakrabarti, S.: Annotating and searching web tables using entities, types and relationships. Proc. VLDB Endow. **3**(1), 1338–1347 (2010)
125. Linked open data cloud (2018). https://www.lod-cloud.net/
126. Liu, B., Jagadish, H.V.: Datalens: making a good first impression. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29–July 2, 2009, pp. 1115–1118 (2009). https://doi.org/10.1145/1559845.1559997
127. Maali, F., Erickson, J., Archer, P.: Data catalog vocabulary (dcat). W3C Recommendation, vol. 16 (2014). https://www.w3.org/TR/vocab-dcat/#class-dataset
128. Madhavan, J., Ko, D., Kot, Ł., Ganapathy, V., Rasmussen, A., Halevy, A.: Google's deep web crawl. Proc. VLDB Endow. **1**(2), 1241–1252 (2008)
129. Madhu, G., Govardhan, D.A., Rajinikanth, D.T.: Intelligent semantic web search engines: a brief survey (2011). arXiv preprint arXiv:1102.0831
130. Marchionini, G., Haas, S.W., Zhang, J., Elsas, J.: Accessing government statistical information. Computer **38**(12), 52–61 (2005). https://doi.org/10.1109/MC.2005.393
131. MELODA: Meloda dataset definition (2018). http://www.meloda.org/dataset-definition/
132. Miao, X., Gao, Y., Guo, S., Liu, W.: Incomplete data management: a survey. Front. Comput. Sci. **12**, 1–22 (2018)
133. Missier, P., M. Embury, S., Mark Greenwood, R., D. Preece, A., Jin, B.: Quality views: capturing and exploiting the user perspective on data quality. In: Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 977–988. VLDB Endowment (2006)
134. Mitra, B., Craswell, N.: Neural models for information retrieval (2017). arXiv preprint arXiv:1705.01509
135. Moreau, L., Groth, P.T.: Provenance: an introduction to PROV. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan and Claypool Publishers (2013). https://doi.org/10.2200/S00528ED1V01Y201308WBE007
136. Mork, P., Smith, K., Blaustein, B., Wolf, C., Samuel, K., Sarver, K., Vayndiner, I.: Facilitating discovery on the private web using dataset digests. Int. J. Metadata Semant. Ontol. **5**(3), 170–183 (2010). https://doi.org/10.1504/IJMSO.2010.034042
137. Naumann, F.: Data profiling revisited. SIGMOD Rec. **42**(4), 40–49 (2014). https://doi.org/10.1145/2590989.2590995
138. Neumaier, S., Polleres, A.: Enabling spatio-temporal search in open data. Tech. rep., Department für Informationsverarbeitung und Prozessmanagement, WU Vienna University of Economics and Business (2018)
139. Neumaier, S., Umbrich, J., Polleres, A.: Automated quality assessment of metadata across open data portals. J. Data Inf. Qual. **8**(1), 2:1–2:39 (2016). https://doi.org/10.1145/2964909
140. Nguyen, T.T., Nguyen, Q.V.H., Weidlich, M., Aberer, K.: Result selection and summarization for web table search. In: 2015 IEEE 31st International Conference on Data Engineering (ICDE), pp. 231–242. IEEE (2015)
141. Noy, N., Burgess, M., Brickley, D.: Google dataset search: building a search engine for datasets in an open web ecosystem. In: 28th Web Conference (WebConf 2019) (2019)
142. Nuzzolese, A.G., Presutti, V., Gangemi, A., Peroni, S., Ciancarini, P.: Aemoo: linked data exploration based on knowledge patterns. Semant. Web **8**(1), 87–112 (2016). https://doi.org/10.3233/SW-160222
143. Oguz, D., Ergenc, B., Yin, S., Dikenelli, O., Hameurlain, A.: Federated query processing on linked data: a qualitative survey and open challenges. Knowl. Eng. Rev. **30**(5), 545–563 (2015)
144. Open data monitor (2018). https://www.opendatamonitor.eu
145. Orr, L., Balazinska, M., Suciu, D.: Probabilistic database summarization for interactive data exploration. Proc. VLDB Endow. **10**(10), 1154–1165 (2017). https://doi.org/10.14778/3115404.3115419
146. Pan, Z., Zhu, T., Liu, H., Ning, H.: A survey of rdf management technologies and benchmark datasets. J. Ambient Intell. Humaniz. Comput. **9**(5), 1693–1704 (2018). https://doi.org/10.1007/s12652-018-0876-2
147. Partnership, O.C.: Open contracting data standard (2015). http://standard.open-contracting.org/latest/en/
148. Pasquetto, I.V., Randles, B.M., Borgman, C.L.: On the reuse of scientific data. Data Sci. J. **16**, 8 (2017). https://doi.org/10.5334/dsj-2017-008
149. Paulheim, H.: Knowledge graph refinement: a survey of approaches and evaluation methods. Semant. Web **8**(3), 489–508 (2016). https://doi.org/10.3233/SW-160218
150. Peng, J., Zhang, D., Wang, J., Pei, J.: AQP++: Connecting approximate query processing with aggregate precomputation for interactive analytics. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, pp. 1477–1492. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3183747
151. Pimplikar, R., Sarawagi, S.: Answering table queries on the web using column keywords. Proc. VLDB Endow. **5**(10), 908–919 (2012). https://doi.org/10.14778/2336664.2336665
152. Pirolli, P., Rao, R.: Table lens as a tool for making sense of data. In: Proceedings of the Workshop on Advanced Visual Interfaces 1996, Gubbio, Italy, May 27–29, 1996, pp. 67–80 (1996). https://doi.org/10.1145/948449.948460
153. Piscopo, A., Phethean, C., Simperl, E.: What makes a good collaborative knowledge graph: group composition and quality in wikidata. In: Ciampaglia, G.L., Mashhadi, A., Yasseri, T. (eds.) Social Informatics, pp. 305–322. Springer, Cham (2017)
154. Rajaraman, A.: Kosmix: high-performance topic exploration using the deep web. Proc. VLDB Endow. **2**(2), 1524–1529 (2009). https://doi.org/10.14778/1687553.1687581
155. Rekatsinas, T., Dong, X.L., Srivastava, D.: Characterizing and selecting fresh data sources. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, SIGMOD '14, pp. 919–930. ACM, New York, USA (2014). https://doi.org/10.1145/2588555.2610504
156. Reynolds, P.: DHS Data Framework DHS/ALL/PIA-046(a). Technical Report, US Department of Homeland Security (2014)
157. Rieh, S.Y., Collins-Thompson, K., Hansen, P., Lee, H.: Towards searching as a learning process: a review of current perspectives and future directions. J. Inf. Sci. **42**(1), 19–34 (2016). https://doi.org/10.1177/0165551515615841
158. Ritze, D., Lehmberg, O., Bizer, C.: Matching HTML tables to dbpedia. In: Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics, WIMS 2015, Larnaca, Cyprus, July 13–15, 2015, pp. 10:1–10:6 (2015). https://doi.org/10.1145/2797115.2797118
159. Roh, Y., Heo, G., Whang, S.E.: A survey on data collection for machine learning: a big data—AI integration perspective (2018). CoRR arXiv:1811.03402
160. Saleem, M., Ngomo, A.N.: Hibiscus: hypergraph-based source selection for SPARQL endpoint federation. In: The Semantic Web: Trends and Challenges—11th International Conference, ESWC 2014, Crete, Greece, May 25–29, 2014. Proceedings, pp. 176–191 (2014). https://doi.org/10.1007/978-3-319-07443-6_13

161. Sansone, S.A., González-Beltrán, A., Rocca-Serra, P., Alter, G., Grethe, J., Xu, H., Fore, I., Lyle, J., E. Gururaj, A., Chen, X., Kim, H., Zong, N., Li, Y., Liu, R., Burak Ozyurt, I., Ohno-Machado, L.: Dats, the data tag suite to enable discoverability of datasets. Sci. Data **4** (2017). https://doi.org/10.1038/sdata.2017.59
162. SDMX: Sdmx glossary. Technical Report, SDMX Statistical Working Group (2018)
163. Search Retrieval via URL: CQL: The contextual query language. The Library of Congress Standards (2016)
164. Shestakov, D., Bhowmick, S.S., Lim, E.P.: Deque: querying the deep web. Data Knowl. Eng. **52**(3), 273–311 (2005). https://doi.org/10.1016/j.datak.2004.06.009
165. Siglmüller, F.: Advanced user interface for artwork search result presentation. Institute of Com (2015)
166. Spiliopoulou, M., Rodrigues, P.P., Menasalvas, E.: Medical mining: Kdd 2015 tutorial. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, pp. 2325–2325. ACM, New York, NY, USA (2015). https://doi.org/10.1145/2783258.2789992
167. Stonebraker, M., Ilyas, I.F.: Data integration: the current status and the way forward. IEEE Data Eng. Bull. **41**(2), 3–9 (2018)
168. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I.J., Fergus, R.: Intriguing properties of neural networks (2013). CoRR arXiv:1312.6199
169. Tang, Y., Wang, H., Zhang, S., Zhang, H., Shi, R.: Efficient web-based data imputation with graph model. In: International Conference on Database Systems for Advanced Applications, pp. 213–226. Springer (2017)
170. Tennison, J.: CSV on the web: a primer. W3C note, W3C (2016). http://www.w3.org/TR/2016/NOTE-tabular-data-primer-20160225/
171. Thelwall, M., Kousha, K.: Figshare: a universal repository for academic resource sharing? Online Inf. Rev. **40**(3), 333–346 (2016). https://doi.org/10.1108/OIR-06-2015-0190
172. Thomas, P., Omari, R.M., Rowlands, T.: Towards searching amongst tables. In: Proceedings of the 20th Australasian Document Computing Symposium, ADCS 2015, Parramatta, NSW, Australia, December 8–9, 2015, pp. 8:1–8:4 (2015). https://doi.org/10.1145/2838931.2838941
173. Townsend, A.: Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia. W.W. Norton and Company, Inc., New York (2013)
174. Uk open data portal (2018). https://data.gov.uk/
175. Umbrich, J., Neumaier, S., Polleres, A.: Quality assessment and evolution of open data portals. In: 2015 3rd International Conference on Future Internet of Things and Cloud, pp. 404–411 (2015). https://doi.org/10.1109/FiCloud.2015.82
176. Van Gysel, C., de Rijke, M., Kanoulas, E.: Neural vector spaces for unsupervised information retrieval. ACM Trans. Inf. Syst. **36**(4), 38 (2018)
177. Vidal, M.E., Castillo, S., Acosta, M., Montoya, G., Palma, G.: On the selection of SPARQL endpoints to efficiently execute federated SPARQL queries. In: Hameurlain, A., Küng, J., Wagner, R. (eds.) Transactions on Large-Scale Data- and Knowledge-Centered Systems. Lecture Notes in Computer Science, vol. XXV, pp. 109–149. Springer, Berlin (2016)
178. W3C: List of known semantic web search engines. https://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/SemanticWebSearchEngines
179. W3C: The rdf data cube vocabulary (2014). https://www.w3.org/TR/vocab-data-cube/t
180. Weerkamp, W., Berendsen, R., Kovachev, B., Meij, E., Balog, K., de Rijke, M.: People searching for people: analysis of a people search engine log. In: Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25–29, 2011, pp. 45–54 (2011). https://doi.org/10.1145/2009916.2009927
181. Wen, Y., Zhu, X., Roy, S., Yang, J.: Interactive summarization and exploration of top aggregate query answers. Proc. VLDB Endow. **11**(13), 2196–2208 (2018). https://doi.org/10.14778/3275366.3275369
182. White, R.W., Bailey, P., Chen, L.: Predicting user interests from contextual information. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19–23, 2009, pp. 363–370 (2009). https://doi.org/10.1145/1571941.1572005
183. Wiggins, A., Young, A., Kenney, M.A.: Exploring visual representations to support datafire-use for interdisciplinary science. Assoc. Inf. Sci. Technol. **55**, 554–563 (2018)
184. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., Bouwman, J., Brookes, A.J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C.T., Finkers, R., Gonzalez-Beltran, A., Gray, A.J., Groth, P., Goble, C., Grethe, J.S., Heringa, J., 't Hoen, P.A., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S.J., Martone, M.E., Mons, A., Packer, A.L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M.A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., Mons, B.: The FAIR Guiding Principles for scientific data management and stewardship. Sci. Data **3**, 160018 (2016). https://doi.org/10.1038/sdata.2016.18
185. Woodall, P., Wainman, A.: Data quality in analytics: key problems arising from the repurposing of manufacturing data. In: Proceedings of the International Conference on Information Quality (2015)
186. Wu, Y., Alawini, A., Davidson, S.B., Silvello, G.: Data citation: giving credit where credit is due. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, pp. 99–114. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3196910
187. Wylot, M., Cudré-Mauroux, P., Hauswirth, M., Groth, P.T.: Storing, tracking, and querying provenance in linked data. IEEE Trans. Knowl. Data Eng. **29**(8), 1751–1764 (2017). https://doi.org/10.1109/TKDE.2017.2690299
188. Wylot, M., Hauswirth, M., Cudré-Mauroux, P., Sakr, S.: RDF data storage and query processing schemes: a survey. ACM Comput. Surv. **51**(4), 84:1–84:36 (2018)
189. Xiao, D., Bashllari, A., Menard, T., Eltabakh, M.: Even metadata is getting big: annotation summarization using insightnotes. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, SIGMOD '15, pp. 1409–1414. ACM, New York, NY, USA (2015). https://doi.org/10.1145/2723372.2735355
190. Yakout, M., Ganjam, K., Chakrabarti, K., Chaudhuri, S.: Infogather: entity augmentation and attribute discovery by holistic matching with web tables. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, SIGMOD '12, pp. 97–108. ACM, New York, NY, USA (2012). https://doi.org/10.1145/2213836.2213848
191. Yan, C., He, Y.: Synthesizing type-detection logic for rich semantic data types using open-source code. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD '18, pp. 35–50. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3183713.3196888
192. Yoghourdjian, V., Archambault, D., Diehl, S., Dwyer, T., Klein, K., Purchase, H.C., Wu, H.Y.: Exploring the limits of complexity: a survey of empirical studies on graph visualisation. Vis. Inform.

2(4), 264–282 (2018). https://doi.org/10.1016/j.visinf.2018.12.006
193. Yoghourdjian, V., Dwyer, T., Klein, K., Marriott, K., Wybrow, M.: Graph thumbnails: identifying and comparing multiple graphs at a glance. IEEE Trans. Vis. Comput. Graph. **24**(12), 3081–3095 (2018). https://doi.org/10.1109/tvcg.2018.2790961
194. Yu, P.S., Li, X., Liu, B.: Adding the temporal dimension to search—a case study in publication search. In: 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005), 19–22 September 2005, Compiegne, France, pp. 543–549 (2005). https://doi.org/10.1109/WI.2005.21
195. Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S.: Quality assessment for linked data: a survey. Semant. Web **7**(1), 63–93 (2016)
196. Zhang, S.: Smarttable: equipping spreadsheets with intelligent assistancefunctionalities. In: The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '18, pp. 1447–1447. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3209978.3210219
197. Zhang, S., Balog, K.: Entitables: smart assistance for entity-focused tables. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7–11, 2017, pp. 255–264 (2017). https://doi.org/10.1145/3077136.3080796
198. Zhang, S., Balog, K.: Ad hoc table retrieval using semantic similarity. In: Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23–27, 2018, pp. 1553–1562 (2018). https://doi.org/10.1145/3178876.3186067
199. Zhang, S., Balog, K.: On-the-fly table generation. In: The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '18, pp. 595–604. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3209978.3209988
200. Zhang, X., Wang, J., Yin, J.: Sapprox: enabling efficient and accurate approximations on sub-datasets with distribution-aware online sampling. Proc. VLDB Endow. **10**(3), 109–120 (2016). https://doi.org/10.14778/3021924.3021928