The Case for Holistic Data Integration

Erhard Rahm(✉)

University of Leipzig, Leipzig, Germany
rahm@informatik.uni-leipzig.de

Abstract. Current data integration approaches are mostly limited to few data sources, partly due to the use of binary match approaches between pairs of sources. We thus advocate for the development of more holistic, clustering-based data integration approaches that scale to many data sources. We outline different use cases and provide an overview of initial approaches for holistic schema/ontology integration and entity clustering. The discussion also considers open data repositories and so-called knowledge graphs.

1 Introduction

Data integration aims at providing uniform access to data from multiple sources [17]. It has become a pervasive task for data analysis in business and scientific applications. The most popular data integration approaches such as data warehouses or big data platforms utilize a physical data integration where the source data is combined within a new dataset or database tailored for analysis tasks. This is in contrast to virtual data integration where data entities remain in their original data sources and are accessed at runtime, e.g., for federated query processing. Federated query processing has also become popular in the so-called Web of Data, also referred to as Linked Open Data (LOD), and is supported by semantic links interconnecting different sources [63,67].

Key tasks for data integration include data preprocessing (data cleaning [62], data enrichment), entity resolution (data matching) [13,20], entity fusion [9], as well as matching and merging metadata models such as schemas and ontologies [7,61]. Data enrichment can often be achieved by linking entities and/or metadata such as attribute names to background knowledge resources (e.g., dictionaries, ontologies, knowledge graphs), which is a non-trivial mapping and data integration problem in itself [68]. The different data integration tasks have been the focus of a huge amount of research and development. Still, the mentioned tasks are inherently complex and are in many cases not performed fully automatically but incur a high degree of manual interaction. This is because data sources may be of low data quality, may be unstructured or follow different data formats (relational, JSON, etc.) and exhibit a high degree of semantic heterogeneity since they are mostly developed independently for different purposes.

These problems increase with the number of data sources to be integrated. As a result, most data integration approaches and efforts focus on only a few data sources. Data matching and schema matching approaches mostly determine

J. Pokorný et al. (Eds.): ADBIS 2016, LNCS 9809, pp. 11–27, 2016.
DOI: 10.1007/978-3-319-44039-2_2

correspondences (links) between only two sources. While pairwise matching is a building block for most data integration solutions, the sole generation of such binary mapping approaches does not scale to many data sources as the number of possible mappings increases quadratically with the number of sources. For example, fully interlinking 200 LOD sources would require the determination and maintenance of almost 20,000 mappings.

We thus see a strong and increasing need for *holistic data integration* approaches that can integrate many data sources. To be scalable, holistic data integration should not be limited to pairwise matching and integration of sources but support a clustering-based integration of both metadata[1] and instance data to holistically combine the information from many sources. The need for such holistic approaches is fueled by the availability of relevant data in millions of websites and the provision of large data and metadata collections in public (open data) repositories. Platforms such as *data.gov*, *www.opensciencedatacloud.org*, *datahub.io* and *webdatacommons.org* contain thousands of datasets and millions of web extractions (e.g., web tables) for many topics in different domains. There are also repositories for metadata (schemas, ontologies) and mappings, e.g., *schema.org*, *medical-data-models.org*, Linked Open Vocabularies (*lov.okfn.org*), BioPortal [52], and LinkLion [49], supporting the re-use of this information to facilitate data integration tasks.

To achieve scalability to many sources, holistic data integration approaches should be fully automatic or require only minimal manual interaction. It should also be easily possible to add and utilize additional data sources and deal with changes in the data sources. As with all data integration approaches, high efficiency and high data integration quality need to be supported which becomes more challenging due to the increased number of (heterogeneous) sources and the typically much increased data volume. High efficiency asks for the utilization of powerful (big data) platforms for parallel processing and blocking-like techniques to reduce the search space for match tasks. Achieving high data integration quality and avoiding/minimizing manual interaction are contradictory goals so that viable compromises need to be found.

The main goal of this paper is to motivate the need for holistic data integration with different use cases and to provide an overview of initial approaches. In Sect. 2, we outline six use cases for holistic integration of metadata or entities. Section 3 discusses approaches to match and merge many schemas and ontologies as well as the use of open data repositories. In Sect. 4, we focus on the holistic clustering of entities of different types, e.g. for LOD sources or to determine knowledge graphs. Finally, we summarize our observations and discuss opportunities for future research.

[1] In this paper, we are only concerned with metadata in the form of schemas and ontologies and their components like attributes or concepts. We are thus not considering the wide range of additional metadata (e.g., provenance information, creator, creation time, etc.) despite their importance, e.g., for data quality.

2 Use Cases

Table 1 lists six examples for holistic data integration together with estimates on the number of domains, the number of sources, features about the kind of data integration (physical vs. virtual), and whether the focus is on data integration for metadata (schemas/ontologies) and/or instance data. We also indicate the kind of clustering and to what degree data integration can likely be automated.

The first two use cases, meta-search and the use of open data, focus on simple schemas such as web forms or tables consisting of relatively few attributes. *Meta-search* is a virtual data integration approach based on metadata integration. The goal is to integrate the search forms of several databases of the so-called hidden web to support a meta-search across all sources, e.g., for comparing products from different online shops. Schema integration mainly entails grouping or clustering similar attributes, which is simpler than matching and merging complex schemas. As a result, scalability to dozens of sources is typically feasible. Proposed approaches include Wise-Integrator and MetaQuerier [12,33].

A completely different situation is when there is an enormous number of datasets such as web tables made available within *open data repositories*. The physically collected datasets are typically from diverse domains and initially not integrated at all. To enable their usability, e.g., for query processing, it is useful to group the datasets into different domains and to semantically annotate attributes. Google Fusion Tables has demonstrated the utilization of millions of such semantically annotated web tables to better answer certain search queries [4]. Semantically enriched attributes could also be used to match and cluster datasets such as web tables within the repository. Problems similar to those for open data repositories arise for so-called "data lake" approaches to collect datasets in their original format for later use [27,55].

Table 1. Use cases for holistic data integration.

Use case	Data integration		#domains	#sources	Clustering?	Degree of automated data integration
(1) Meta-search	Virtual	Metadata	1	Low - medium	Attributes	Medium
(2) Open data	Physical collection	Primarily metadata	Many	Very high	(Possible)	High, but limited integration
(3) Integrated ontology	Physical	Metadata	1+	Low - medium	Concepts	Low - medium
(4) Knowledge graphs	Physical	Data + metadata	Many	Low - high	Entities + concepts/ attributes	Medium - high
(5) Entity search engines	Physical	Data (+ metadata)	1	Very high	Entities	High
(6) Comparison portal	Physical/ hybrid	Data + metadata	1+	High	Entities	High

The next two use cases are concerned with physical data integration to determine integrated background knowledge resources such as large domain ontologies or multi-domain knowledge graphs. In the first case (use case 3) the goal is to semantically merge several related ontologies into a combined ontology to consistently represent the knowledge of a domain. This implies the identification of synonymous concepts across all source ontologies as well as the derivation of a consistent ontology structure for these concepts and their relations. An example of such an integration effort is the biomedical ontology UMLS Metathesaurus [10] which currently (2016) combines more than three million concepts and more than 12 million synonyms from more than 100 biomedical ontologies and vocabularies. The integration process is highly complex and involves a significant effort by domain experts. Another example for holistic metadata integration is the construction of an integrated product catalog from several merchant-specific catalogs, e.g., for price comparisons.

The generation of so-called *knowledge graphs* [18] is a related use case for holistic data integration where concepts as well as entities from different sources are physically integrated. Popular knowledge graphs in the Web of Data are DBpedia, Yago and Wikidata [3,41,70,73] that extract information about millions of real-world entities (such as persons or locations) of different domains as well as concepts from other resources such as Wikipedia or WordNet. The entities are placed within a categorization or class (concept) hierarchy and interlinked with a variety of semantic relationships. Web search engines such as Google or Bing utilize even larger knowledge graphs [51] combining information from additional resources as well as from web pages and search queries. Knowledge graphs can provide valuable background knowledge, e.g., to enrich entities mentioned in text documents or to enhance the search results for web queries. Web-scale knowledge graphs for many domains ask for highly automated data integration methods but face substantial challenges regarding data quality and semantic heterogeneity [18,26]. So-called enterprise knowledge graphs focus on the datasets relevant for an enterprise and their semantic integration [22].

Entity search engines such as Google Scholar or Bing Shopping (use case 5) cluster corresponding entities such as publication records or product offers from thousands to millions of data sources or web pages. The focus is on physical clustering at the instance level. The quality and usability of clustering can be improved by assigning the entities to categories, e.g., for products, which may be arranged in a product catalog, e.g., organized as a hierarchical taxonomy. *Comparison portals* for hotel bookings, product offers, etc. (use case 6) are similar to entity search engines in that they cluster comparable offers for the same product or booking request. They are typically more selective in the sources they include and may obtain their data in curated form rather than by extracting the entities from web pages as in the case of Google Scholar. Data integration is mostly physical but may also be virtual to retrieve the most recent information, e.g., about the availability of bookable items such as flight seats or hotel rooms. Furthermore, the categorization of entities along different dimensions is the norm to enhance the browsing and search facilities for portal users. This kind of use case involves highly challenging data integration problems, in particular to automatically cluster a huge number of continuously updated product offers from many sources within thousands of product categories described by different sets of attributes and schemas [54].

The discussed use cases show that holistic data integration has wide applicability with significant differences in the considered characteristics. All use cases with a large number of sources utilize physical data integration and are primarily focused on instance-level integration based on a clustering of matching entities. By contrast, metadata integration is limited to a small to medium number of sources and depends more on manual interaction to deal with the typically high complexity. Holistic metadata integration can utilize a clustering of concept synonyms as well as a clustering of attributes per concept or entity type. Virtual data integration generally depends on metadata integration and is thus of limited scalability for complex sources. Scalability of virtual integration is also impaired by likely performance problems for queries involving many sources that typically differ in their capacity, utilization and availability.

3 Holistic Integration of Schemas and Ontologies

Most work on the integration of schemas and ontologies has focused on the pairwise matching of such models, i.e., determining semantically corresponding elements such as pairs of matching schema attributes or ontology concepts [7,21,61]. Matches are usually identified by a combination of techniques to determine the similarity of elements. This includes 1. the linguistic similarity of element names (based on string similarity measures or synonym information from background knowledge resources such as dictionaries), 2. the structural similarity of elements (e.g., based on the similarity of ancestors and/or descendants) and 3. the similarity of associated instance data. The set of determined match correspondences forms a *mapping* between the two aligned schemas/ontologies. Such match mappings are useful input to merge or integrate the respective models since they indicate the elements that should only be represented once in the integrated result. In fact, several such mapping-based merge approaches have been proposed for both schemas [58,59] and ontologies [64].

In the following, we first discuss proposed holistic match and merge approaches for complex schemas and ontologies, including for LOD sources. Afterwards we discuss proposed data integration approaches for simple schemas such as web forms and web tables.

Complex Schemas and Ontologies. In principle, the pairwise matching and merging can be applied to more than two models by incrementally matching and merging two models at a time. For instance, one can use one of the schemas as the initial integrated schema and incrementally match and merge the next source with the intermediate result until all source schemas are integrated. Such a binary integration strategy for multiple schemas has already been considered in early work on schema integration [6], however based on a largely manual process. More recently it has been applied within the Porsche approach [66] to automatically merge many tree-structured XML schemas. The approach holistically clusters all matching elements in the nodes of the integrated schema. The placement of new source elements not found in the (intermediate) integrated schema is based on a simplistic heuristic only. A general problem of incremental merge approaches is that the final merge result depends on the order in which the input schemas are matched and merged.

The matching between many schemas and ontologies can be facilitated by the re-use of previously determined mappings between such models, especially if such mappings are available in repositories like Bio-Portal [52]. Such a re-use of mappings has already been proposed in the 2001 survey [61] and several match approaches are utilizing re-use techniques based on a repository of schemas and mappings [16,43,65]. A simple and effective approach is based on the composition of existing mappings to quickly derive new mappings. In particular, one can derive a new mapping between schemas S_1 and S_2 by composing existing mappings, e.g., mappings between $S_1$and S_i and between S_i and S_2 for any intermediate schema S_i (Fig. 1 left). Such composition approaches have been investigated in [23,28] and were shown to be very fast and also effective, especially if one can combine several such derived mappings for improved coverage of the schemas to be matched. A promising strategy is to utilize a hub schema (ontology) per domain to which all other schemas are mapped. Then one can derive a mapping between any two schemas by composing their mappings with the hub schema (Fig. 1 right).

The next step would be to integrate all schemas with the hub schema together with a clustering of the matching elements. Such integrated hub ontologies have been determined in the life sciences, e.g., UMLS [10] and Uberon [45], although with the need of a large amount of manual work by domain experts to achieve a high-quality integration result. A more automatic integration becomes feasible for the integration of simpler ontologies such as dictionaries or thesauri. An example is the SemRep repository [2] combining millions of concepts and semantic relations (equal, is-a, part-of, etc.) between them extracted from Wikipedia as well as obtained from existing resources such as WordNet.

Pairwise matching has been applied in [35] to match the terms of more than 4000 web-extracted ontologies (including large LOD sources such as DBpedia) with a total of more than 2 million terms. The match process using a state-of-the-art match tool took about one year on six computers showing the insufficient scalability of pairwise matching. A holistic matching of concepts in LOD sources has been proposed in [25]. The authors first cluster the concepts within different topical groups and then apply pairwise matching of concepts within groups to finally determine clusters of matching concepts. For clustering and matching they derive keywords from the concept labels and descriptions, determine associated (trees of) categories in Wikipedia and use these to derive concept similarities (similarly as for the BLOOMS match technique [36]). In the evaluation, the authors originally considered 1 million concepts from which less than 30 % could be annotated with Wikipedia categories. Topical grouping was then possible for 162 K concepts (using the preferred configuration) that were assigned to about 32 K groups with a maximal size of about 5 K concepts. Matching for

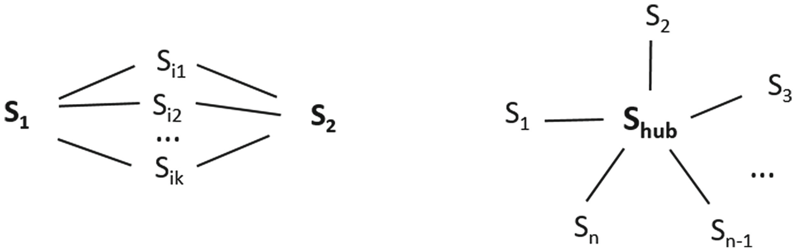

Fig. 1. Composition of mappings to match many schemas

the largest group took more than 30 h. The approach is an interesting first step but it requires improved scalability and coverage, e.g., by applying additional match techniques than the use of Wikipedia categories. Furthermore, clustering is needed not only for concepts but also for LOD entities (Sect. 4).

Simple Schemas. The holistic integration of many schemas has mainly been studied for simple schemas such as web forms and web tables (use cases 1 and 2). As we will discuss in the following, previous work for web forms focused on their integration within a mediated schema as well as on their categorization into different domains. For web tables, the focus has been on the semantic annotation and matching of attributes.

The integration of web forms has been studied to support a meta-search across deep web sources [12,33]. Schema integration implies clustering all similar attributes from the web forms, mainly based on the linguistic similarity of the attribute names (labels) [60]. The approaches also observe that similarly named attributes co-occuring in the same schema (e.g., *FirstName* and *LastName*) do not match and should not be clustered together [31]. Das Sarma and colleagues propose the automatic generation of a so-called probabilistic mediated schema from n input schemas, which is in effect a ranked list of several mediated schemas [14]. Their proposed approach only considers the more frequently occurring attributes and uses their pairwise similarities for determining the different mediated schemas.

The holistic integration of several schemas is generally only relevant for schemas of the same application domain. For a very large number of schemas, it is thus important to first categorize schemas by domain. Several approaches have been proposed for the automatic domain categorization problem of web forms [5,32,44], typically based on a clustering of attribute names and the use of further features such as explaining text in the web page where the form is placed. While approaches such as [5,32] considered the domain categorization for only few predefined domains, Mahmnoud and Aboulnaga [44] cluster schemas into a previously unknown number of domain-like groups that may overlap. In [19], this approach has also been applied for a domain categorization of web tables from a large corpus.

For huge collections of web tables the domain categorization is especially important but cannot successfully be accomplished by only considering attribute names which are often cryptic or very general. This is also a problem for further tasks such as finding related web tables (e.g., to answer queries or to extend web tables with additional attributes) or matching attributes within a corpus of web tables. Hence, it is necessary to consider additional information such as the attribute (instance) values in tables as well as information from the table context in the web pages [4]. Furthermore, it is necessary to semantically enrich attribute information by utilizing external background information such as knowledge graphs, in particular to determine the semantic data type or concept classes of attributes, e.g., company, politician, date-of-birth, country, capital, population etc. Also, relationships between attributes of the same table should be identified. Such semantic enrichment approaches have been investigated in [15,30,42,72,74] utilizing different knowledge resources such as Yago, DBpedia, or Probase. In [72], Google researchers utilized web-crawled knowledge of about 60,000 classes with at least 10 associated entities to find about 1.5 million "subject" attributes in a web table corpus (about 8 times more than using the Wikipedia-based Yago knowledge base).

The Infogather system [76] utilizes such enriched attribute information to match web tables with each other. To limit the scope they determine topic-specific schema match graphs that only consider schemas similar to a specific query table. The match graphs help to determine matching tables upfront before query answering and to holistically utilize information from matching tables. Instance-based approaches to match the attributes of web tables considering the degree of overlap in the attribute values have been used in [19].

Despite such approaches the information in open data repositories is not yet sufficiently utilized. Attribute matching could be improved by considering both, attribute metadata and instances, not just one of them. Further approaches could apply physical data integration, e.g., to combine and cluster matching entities from different tables or to extract entities to build or extend domain-specific knowledge graphs.

4 Holistic Integration of Entities

Entity resolution (also called deduplication, object matching or link discovery) [13,20] has mostly been investigated for finding matching entities[2] (e.g. persons, products, publications, and movies) within a single source or between two sources. For a single source, matching entities are typically grouped within disjoint clusters such that any two entities in a cluster should match with each other and no entity should match with entities of other clusters. For two sources, the match result is mostly a binary mapping consisting of pairs of matching entities (also called match correspondences or links). Binary match mappings may be postprocessed to determine clusters of matching entities, e.g., by calculating the transitive closure of the correspondences and refining the resulting connected components (clusters) to ensure that indirectly linked entities are

[2] To be more precise, we can only find matching records referring to the same real-word object. For simplification, we use the term "entity" to refer to both the records as well as the real-world objects they describe.

really similar enough to stay in the same cluster [29,34,46]. Alternatively, one can construct a similarity graph from the match correspondences and determine subgraph clusters of connected and highly similar entities [24,57].

The match decision is typically based on the combined similarity of several attribute values and possibly on the contextual similarity of entities. In current systems, the combination of the similarity values for deriving a match decision is either based on supervised classification models (learned from training examples) or on manually determined match rules [38,48]. To achieve high efficiency for large datasets, one has to avoid comparing each entity to all other entities. This is made possible by utilizing so-called blocking strategies [13,53,75] and additional filter techniques tailored to specific similarity or distance functions (e.g., the triangle inequality for metric-space distance functions) [50]. Entity resolution can also be performed in parallel on multiple processors and computing nodes, e.g., on Hadoop platforms [37], to achieve additional performance improvements.

In the following, we first outline a general approach to holistically cluster entities from many sources. We then discuss the use of such an approach for LOD sources as well as for use cases of Sect. 2. Finally, we briefly discuss the integration of entities into knowledge graphs.

Holistic Clustering of Entities. To holistically match entities from many sources, the prevalent approaches for pairwise matching, e.g., within the Web of Data, are no longer sufficient and viable. This is because one would need up to $\frac{n\cdot(n-1)}{2}$ binary match mappings for n data sources, i.e., up to 190 and 19,900 mappings for 20 and 200 sources, respectively. Since each mapping is already expensive to determine for large datasets, it is obvious that the computational effort to determine the mentioned number of mappings is infeasible for a large number of sources. Holistic entity resolution thus should be clustering-based by holistically determining match clusters such that all matching entities from any source are combined in a single cluster. For n duplicate-free sources the size of such a match cluster is limited to at most n entities. Each cluster of $k \leq n$ entities represents $\frac{k\cdot(k-1)}{2}$ match pairs and is thus a much more compact representation than with the use of correspondences. The entities of a cluster should have common attributes to determine the entity similarity but can also have different additional attributes that complement each other. By combining the different attributes of the entities in a cluster within a fused entity it is possible to enrich the entity information across all sources as desirable for data integration. The fused entity can serve as a *cluster representative* that is used to match against further entities.

Clustering the entities across all sources can be performed with much less effort than with determining the quadratic number of binary mappings. For static sources, one can bootstrap the clustering process with one of the sources, e.g., the largest one or a source with known high data quality, and use each of its entities as an initial cluster (assuming duplicate-free sources). Then one matches the entities of one source after another with the cluster representatives to decide on the best-matching cluster or whether an entity should form a new cluster. This process can be continued until all sources are matched and clustered. For

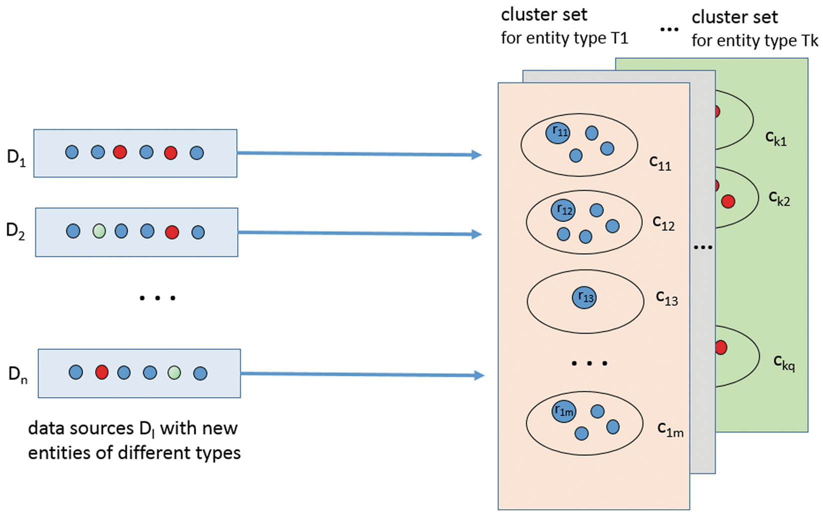

Fig. 2. Holistic clustering of matching entities from multiple sources (clusters are grouped by entity type and have a representative, e.g., r_{ij} for cluster c_{ij} of type T_i)

any entity of any source but the first, the number of match computations is restricted by the number of clusters, which is limited by the total number of distinct entities across all sources. The number of clusters to be considered can be reduced by blocking techniques [13]. In particular, only entities of the same semantic type or class need to be compared with each other, i.e. one should maintain a separate set of clusters for every entity type. Once the entity clusters are established it is relatively easy to match and add new entities from any source, e.g., in a streaming-like manner. Figure 2 illustrates this process where new entities of different types T_i from different sources D_l are matched with the centrally maintained clusters (specifically with cluster representatives r_{ij}) for this entity type. The entity type and other entity attributes may have to be determined during a preprocessing step before the actual match and clustering can begin.

Holistic Clustering of LOD Entities. A holistic clustering of entities is especially promising for LOD data integration which so far is solely based on the use of binary mappings, mostly of type `owl:sameAs` [48]. While a large number of such mappings has already been determined by different tools, the degree of entity linking is still small. One step to improve the situation is to provide predetermined mappings within repositories such as LinkLion [49], and utilize these mappings for deriving additional mappings, e.g., by their transitive composition as used in [11,28]. However, this approach is not sufficient given the large number of LOD sources. Furthermore, existing mappings determined by automatic tools are noisy so that their transitive composition can easily lead to mappings of low quality.

Fortunately, it is possible to apply the sketched holistic entity clustering for LOD sources, as recently proposed in [47]. The approach utilizes existing mappings between n sources of a certain domain, e.g., geographical entities, to determine the transitive closure between them and to postprocess these clusters to ensure a high cluster quality. The approach distinguishes multiple entity types, e.g. cities, mountains, lakes, etc. The entity types provided by the sources are heterogeneous and have to be unified during preprocessing using a predefined type mapping. Unfortunately, for many entities the type is not provided so that it could happen that such untyped entities are clustered with entities of a different type. Furthermore, errors in the input mappings can also lead to wrong entity clusters. For these reasons, the approach postprocesses initially determined clusters to split them to obtain clusters with highly similar entities of the same type. An iterative merge process is also applied to allow entities that have been separated due to a cluster split can be merged with other clusters. The evaluation results showed that the approach clusters many previously unconnected entities thereby resulting in a significantly improved degree of data integration. Furthermore, many errors in the existing mappings could be eliminated, especially by utilizing the type information, e.g., to separate entities with the same names but different types (e.g., city vs. lake).

Further Use Cases. Holistic entity clustering can also be applied for use cases 5 and 6 of Sect. 2, e.g., to cluster publications or product offers. All such use cases require extensive data preprocessing and cleaning to consolidate the entities for matching and also to determine their semantic type since most sources contain different kinds of entities. This is especially the case for product offers, making the operation of a comprehensive price comparison site a highly challenging task. This is because there are typically thousands of product categories each described by different schemas and sets of attributes. Furthermore, there are millions of products offered in thousands of online stores. In addition, product offers change continually (especially on price) and the structure of offers and the attribute values may vary substantially between merchants even for the same product. To facilitate the continuous integration of changing product offers it is important to separate the different product categories and maintain clusters of product offers separately per product type. Product offers should ideally be matched with clean product descriptions serving as cluster representatives. Before new product offers can be matched it is first necessary to determine their product category which can be supported by supervised classification approaches [71]. Furthermore, it is often necessary to extract match-relevant features from text attributes in product offers (e.g., about the manufacturer), to resolve abbreviations and to perform further data cleaning [1]. Matching can then be restricted to the product offers of the selected category and should be based on category-specific match criteria, e.g., category-specific learned classification models [39].

Knowledge Graphs. The generation and continuous refinement of large-scale *knowledge graphs* (use case 4) has similarities to the discussed maintenance of product entities and offers within a large set of heterogeneous product categories. Knowledge graphs typically cover many domains and integrate entities and concepts extracted from Wikipedia, web pages, web search queries and other knowledge resources such as domain ontologies, thesauri etc. [69]. Each entity is typically classified within a large category system and interrelated with other entities. Entities typically have a large number of attributes and attribute values collected and clustered from the different sources [26]. Furthermore, it is desirable to keep track of entity changes over time so that historical versions of entities can be provided [8]. In 2012, the Google knowledge graph contained already 570 million entities within 1500 entity types and 18 billion facts (attribute values, relations) [18]. However, the majority of the automatically collected information is error-prone [18] so that the overall data quality in web-scale knowledge graphs is a massive problem.

To integrate new entities and achieve good data quality, one needs approaches similar to the integration of product offers (categorization of entities, error detection, consolidation of attribute values, entity resolution, etc.), however, they should be able to deal with an even greater scope and diversity of entities. Bellare et al. discuss in [8] the construction of the Yahoo! knowledge graph utilizing a Hadoop infrastructure; entity resolution is based on blocking and pairwise matching followed by a postprocessing to generate entity clusters. Data integration for knowledge graphs also requires the determination and continuous evolution of a fine-grained category system which so far has been largely based on manual decisions. Several studies have begun to address the data quality problems for knowledge graphs, in particular by verifying entity information from multiple sources [18,40]. Paulheim discusses such recent approaches to refine knowledge graphs in [56].

5 Conclusions and Outlook

Traditional data integration approaches that focus on few data sources need to be extended substantially to holistically integrate many sources. In particular, the prevalent pairwise matching of schemas and entities is not scalable enough. The discussion of several use cases and current solutions indicates that holistic data integration should be based on physical data integration as well as on the use of clustering-based approaches to match entities and metadata (concepts, attributes). Scalability for metadata integration is inherently complex and best achieved for simple schemas such as web forms or web tables utilizing a clustering of attributes. Even in this case it is important to utilize large background knowledge resources to semantically categorize and enrich attributes to facilitate data integration. For holistic entity resolution we proposed a general clustering strategy differentiating multiple entity types. Such a scheme can be utilized for a holistic integration of LOD sources as well as for other use cases, e.g., to integrate product offers from numerous online stores. The determination and maintenance of knowledge graphs is especially challenging as it implies the integration of an extremely large number of entities within a huge number of categories. In virtually all use cases, an extensive preprocessing of entities to consolidate and categorize them is of paramount importance for their subsequent integration and use. To limit the amount of manual work for holistic data integration, it seems crucial to build up and re-use curated dictionaries (e.g., to resolve synonyms and abbreviations), schema/ontology and mapping repositories.

The discussion has shown that there are many opportunities to develop new or improved approaches for the holistic integration of metadata and instance data. Open data collections need much more data integration to make them usable, e.g. by categorizing their datasets, clustering entities or deriving domain-specific knowledge graphs. The initial approaches for LOD need to be extended to achieve holistic data integration for both metadata and entities. The approaches for generating and using knowledge graphs need further improvements and evaluation, in particular for largely automatic holistic metadata integration as well as for achieving high data quality. Furthermore, there is a growing need to support fast, near real-time integration of updates and new entities from different sources and data streams. Lastly, scalability techniques including the use of parallel infrastructures and blocking need to be extended to meet the increased performance requirements for holistic data integration.

Acknowledgments. I'd like to thank Sören Auer, Phil Bernstein, Peter Christen, Victor Christen, Anika Groß, Sebastian Hellmann, Dinusha Vatsalan, Qing Wang and Gerhard Weikum for helpful comments and feedback on an earlier version of this paper.

References

1. Arasu, A., Chaudhuri, S., Chen, Z., Ganjam, K., Kaushik, R., Narasayya, V.R.: Experiences with using data cleaning technology for Bing services. IEEE Data Eng. Bull. **35**(2), 14–23 (2012)
2. Arnold, P., Rahm, E.: SemRep: A repository for semantic mapping. In: Proceedings of the BTW, pp. 177–194 (2015)
3. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.G.: DBpedia: A nucleus for a web of open data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007)
4. Balakrishnan, S., Halevy, A.Y., Harb, B., Lee, H., Madhavan, J., Rostamizadeh, A., Shen, W., Wilder, K., Wu, F., Yu, C.: Applying web tables in practice. In: Proceedings of the CIDR (2015)
5. Barbosa, L., Freire, J., Silva, A.: Organizing hidden-web databases by clustering visible web documents. In: Proceedings of the ICDE, pp. 326–335 (2007)
6. Batini, C., Lenzerini, M., Navathe, S.B.: A comparative analysis of methodologies for database schema integration. ACM Comput. Surv. **18**(4), 323–364 (1986)
7. Bellahsene, Z., Bonifati, A., Rahm, E. (eds.): Schema Matching and Mapping. Data-Centric Systems and Applications. Springer, Heidelberg (2011)
8. Bellare, K., Curino, C., Machanavajihala, A., Mika, P., Rahurkar, M., Sane, A.: WOO: A scalable and multi-tenant platform for continuous knowledge base synthesis. PVLDB **6**(11), 1114–1125 (2013)
9. Bleiholder, J., Naumann, F.: Data fusion. ACM Comput. Surv. **41**(1), 1 (2009)
10. Bodenreider, O.: The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res. **32**(suppl 1), D267–D270 (2004)

11. Böhm, C., de Melo, G., Naumann, F., Weikum, G.: LINDA: distributed Web-of-Data-scale entity matching. In: Proceedings of the CIKM, pp. 2104–2108 (2012)
12. Chang, K.C.-C., He, B., Zhang, Z.: Toward large scale integration: Building a MetaQuerier over databases on the web. In: Proceedings of the CIDR (2005)
13. Christen, P.: Data Matching - Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection. Springer, Heidelberg (2012)
14. Sarma, A.D. Dong, X., Halevy, A.: Bootstrapping pay-as-you-go data integration systems. In: Proceedings of the SIGMOD, pp. 861–874 (2008)
15. Deng, D., Jiang, Y., Li, G., Li, J., Yu, C.: Scalable column concept determination for web tables using large knowledge bases. PVLDB **6**(13), 1606–1617 (2013)
16. Do, H.-H., Rahm, E.: COMA: A system for flexible combination of schema matching approaches. In: Proceedings of the VLDB, pp. 610–621 (2002)
17. Doan, A., Halevy, A.Y., Ives, Z.G.: Principles of Data Integration. Morgan Kaufmann, San Francisco (2012)
18. Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., Zhang, W.: Knowledge Vault: A web-scale approach to probabilistic knowledge fusion. In: Proceedings of the SIGKDD, pp. 601–610 (2014)
19. Eberius, J., Damme, P., Braunschweig, K., Thiele, M., Lehner, W.: Publish-time data integration for open data platforms. In: Proceedings of the ACM Workshop on Open Data (2013)
20. Elmagarmid, A.K., Ipeirotis, P.G., Verykios, V.S.: Duplicate record detection: A survey. IEEE TKDE **19**(1), 1–16 (2007)
21. Euzenat, J., Shvaiko, P., et al.: Ontology Matching. Springer, Heidelberg (2007)
22. Galkin, M., Auer, S., Scerri, S.: Enterprise knowledge graphs: A survey. Technical report (2016). http://www.researchgate.net
23. Gross, A., Hartung, M., Kirsten, T., Rahm, E.: Mapping composition for matching large life science ontologies. In: Proceedings of the ICBO (2011)
24. Gruenheid, A., Dong, X.L., Srivastava, D.: Incremental record linkage. PVLDB **7**(9), 697–708 (2014)
25. Gruetze, T., Böhm, C., Naumann, F.: Holistic and scalable ontology alignment for linked open data. In: Proceedings of the LDOW (2012)
26. Gupta, R., Halevy, A., Wang, X., Whang, S.E., Wu, F.: Biperpedia: An ontology for search applications. PVLDB **7**(7), 505–516 (2014)
27. Hai, R., Geisler, S., Quix, C.: Constance: An intelligent data lake system. In: Proceedings of the SIGMOD (2016)
28. Hartung, M., Groß, A., Rahm, E.: Composition methods for link discovery. In: Proceedings of the BTW Conference (2013)
29. Hassanzadeh, O., Chiang, F., Lee, H.C., Miller, R.J.: Framework for evaluating clustering algorithms in duplicate detection. PVLDB **2**(1), 1282–1293 (2009)
30. Hassanzadeh, O., Ward, M.J., Rodriguez-Muro, M., Srinivas, K.: Understanding a large corpus of web tables through matching with knowledge bases-an empirical study. In: Proceedings of the Ontology Matching Workshop (2015)
31. He, B., Chang, K.C.-C.: Statistical schema matching across web query interfaces. In: Proceedings of the SIGMOD, pp. 217–228 (2003)
32. He, B., Tao, T., Chang, KC.-C.: Organizing structured web sources by query schemas: A clustering approach. In: Proceedings of the CIKM, pp. 22–31 (2004)
33. He, H., Meng, W., Yu, C., Wu, Z.: WISE-Integrator: An automatic integrator of web search interfaces for E-commerce. In: Proceedings of the 29th VLDB Conference (2003)
34. Hernández, M.A., Stolfo, S.J.: The merge/purge problem for large databases. ACM SIGMOD Rec. **24**(2), 127–138 (1995)

35. Hu, W., Chen, J., Zhang, H., Qu, Y.: How matchable are four thousand ontologies on the semantic web. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) ESWC 2011, Part I. LNCS, vol. 6643, pp. 290–304. Springer, Heidelberg (2011)
36. Jain, P., Hitzler, P., Sheth, A.P., Verma, K., Yeh, P.Z.: Ontology alignment for linked open data. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part I. LNCS, vol. 6496, pp. 402–417. Springer, Heidelberg (2010)
37. Kolb, L., Thor, A., Rahm, E.: Dedoop: Efficient deduplication with hadoop. PVLDB **5**(12), 1878–1881 (2012)
38. Köpcke, H., Rahm, E.: Frameworks for entity matching: A comparison. Data Knowl. Eng. **69**(2), 197–210 (2010)
39. Köpcke, H., Thor, A., Thomas, S., Rahm, E.: Tailoring entity resolution for matching product offers. In: Proceedings of the EDBT, pp. 545–550 (2012)
40. Lee, T., Wang, Z., Wang, H., Hwang, S.-W.: Web scale taxonomy cleansing. PVLDB **4**(12), 1295–1306 (2011)
41. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., et al.: DBpedia-a large-scale, multilingual knowledge base extracted from Wikipedia. Semant. Web J. **6**(2), 167–195 (2015)
42. Limaye, G., Sarawagi, S., Chakrabarti, S.: Annotating and searching web tables using entities, types and relationships. PVLDB **3**(1–2), 1338–1347 (2010)
43. Madhavan, J., Bernstein, P.A., Doan, A., Halevy, A.: Corpus-based schema matching. In: ICDE, pp. 57–68 (2005)
44. Mahmoud, H.A., Aboulnaga, A.: Schema clustering and retrieval for multi-domain pay-as-you-go data integration systems. In: Proceedings of the SIGMOD (2010)
45. Mungall, C.J., Torniai, C., Gkoutos, G.V., Lewis, S.E., Haendel, M.A., et al.: Uberon, an integrative multi-species anatomy ontology. Genome Biol. **13**(1), R5 (2012)
46. Naumann, F., Herschel, M.: An introduction to duplicate detection. Synthesis Lectures on Data Management **2**(1), 1–87 (2010)
47. Nentwig, M., Groß, A., Rahm, E.: Holistic entity clustering for linked data. University of Leipzig, Technical report (2016)
48. Nentwig, M. Hartung, M., Ngomo, A.-C.N., Rahm, E.: A survey of current link discovery frameworks. Semant. Web J. (2016)
49. Nentwig, M., Soru, T., Ngomo, A.-C.N., Rahm, E.: LinkLion: A link repository for the web of data. In: Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A. (eds.) ESWC Satellite Events 2014. LNCS, vol. 8798, pp. 439–443. Springer, Heidelberg (2014)
50. Ngomo, A.-C.N., Auer, S.: LIMES - A time-efficient approach for large-scale link discovery on the web of data. In: Proceedings of the IJCAI, pp. 2312–2317 (2011)
51. Nickel, M., Murphy, K., Tresp, V., Gabrilovich, E.: A review of relational machine learning for knowledge graphs. Proc. IEEE **104**(1), 11–33 (2016)
52. Noy, N., et al.: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Res. **37**, W170–W173 (2009)
53. Papadakis, G., Ioannou, E., Niederée, C., Palpanas, T., Nejdl, W.: Beyond 100 million entities: large-scale blocking-based resolution for heterogeneous data. In: Proceedings of the ACM Conference Web search and data mining, pp. 53–62 (2012)

54. Papadimitriou, P., Tsaparas, P., Fuxman, A., Getoor, L.: TACI: Taxonomy-aware catalog integration. IEEE TKDE **25**(7), 1643–1655 (2013)
55. Pasupuleti, P., Purra, B.S.: Data Lake Development with Big Data. Packt Publishing Ltd., Birmingham (2015)
56. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semant. Web J. (2016)
57. Pershina, M., Yakout, M., Chakrabarti, K.: Holistic entity matching across knowledge graphs. In: IEEE International Conference on Big Data, pp. 1585–1590 (2015)
58. Pottinger, R.A., Bernstein, P.A.: Merging models based on given correspondences. In: Proceedings of the VLDB, pp. 862–873 (2003)
59. Radwan, A., Popa, L., Stanoi, I.R., Younis, A.: Top-k generation of integrated schemas based on directed and weighted correspondences. In: Proceedings of the SIGMOD, pp. 641–654 (2009)
60. Rahm, E.: Towards large-scale schema and ontology matching. In: Bellahsene, Z., Bonifati, A., Rahm, E. (eds.) Schema Matching and Mapping. Data-Centric Systems and Applications, pp. 3–27. Springer, Heidelberg (2011)
61. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. VLDB J. **10**, 334–350 (2001)
62. Rahm, E., Do, H.H.: Data cleaning: Problems and current approaches. IEEE Data Eng. Bull. **23**(4), 3–13 (2000)
63. Rakhmawati, N.A., Umbrich, J., Karnstedt, M., Hasnain, A., Hausenblas, M.: A Comparison of Federation over SPARQL Endpoints Frameworks. In: Klinov, P., Mouromtsev, D. (eds.) KESW 2013. CCIS, vol. 394, pp. 132–146. Springer, Heidelberg (2013)
64. Raunich, S., Rahm, E.: Target-driven merging of taxonomies with ATOM. Inf. Syst. **42**, 1–14 (2014)
65. Saha, B., Stanoi, I., Clarkson, K.L.: Schema covering: a step towards enabling reuse in information integration. In: ICDE, pp. 285–296 (2010)
66. Saleem, K., Bellahsene, Z., Hunt, E.: Porsche: Performance oriented schema mediation. Inf. Syst. **33**(7), 637–657 (2008)
67. Schwarte, A., Haase, P., Hose, K., Schenkel, R., Schmidt, M.: FedX: Optimization techniques for federated query processing on linked data. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) ISWC 2011, Part I. LNCS, vol. 7031, pp. 601–616. Springer, Heidelberg (2011)
68. Shen, W., Wang, J., Han, J.: Entity linking with a knowledge base: Issues, techniques, and solutions. IEEE TKDE **27**(2), 443–460 (2015)
69. Suchanek, F., Weikum, G.: Knowledge harvesting in the big-data era. In: Proceedings of the SIGMOD, pp. 933–938 (2013)
70. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: A large ontology from wikipedia and wordnet. Web Semant. Sci. Serv. Agents World Wide Web **6**(3), 203–217 (2008)
71. Sun, C., Rampalli, N., Yang, F., Doan, A.: Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. PVLDB **7**(13), 1529–1540 (2014)
72. Venetis, P., Halevy, A., Madhavan, J., Paşca, M., Shen, W., Wu, F., Miao, G., Wu, C.: Recovering semantics of tables on the web. PVLDB **4**(9), 528–538 (2011)
73. Vrandečić, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. CACM **57**(10), 78–85 (2014)

74. Wang, J., Wang, H., Wang, Z., Zhu, K.Q.: Understanding tables on the web. In: Atzeni, P., Cheung, D., Ram, S. (eds.) ER 2012 Main Conference 2012. LNCS, vol. 7532, pp. 141–155. Springer, Heidelberg (2012)
75. Whang, S.E., Menestrina, D., Koutrika, G., Theobald, M., Garcia-Molina, H.: Entity resolution with iterative blocking. In: Proceedings of the SIGMOD, pp. 219–232 (2009)
76. Yakout, M., Ganjam, K., Chakrabarti, K., Chaudhuri, S.: Infogather: entity augmentation and attribute discovery by holistic matching with web tables. In: Proceedings of the SIGMOD, pp. 97–108, (2012)