Challenges in the Design, Implementation, Operation and Maintenance of Knowledge Graphs

Gary Berg-Cross

RDA/US Advisory Group

Abstract

As a useful information system product, a knowledge graph (KG) must be assembled from many diverse, independently developed sources of information. Sources range from simple text and textual definitions to formal ontologies. Along the path of constructing a KG there are many challenges in the design, assembly and implementation. Data and knowledge challenges, including semantic ones, exist at every step of KG lifecycle processes. These include many recursive steps to align, refine and validate the information product. Internal data management problems such as entity identification and refinement are mixed in with external challenges such as the useful scoping of information. And there are sociotechnical challenges as well, such as the best use of interdisciplinary teams. This article is an attempt to overview some of the issues discussed at the 2020 Ontology Summit and related literature on these challenges.

Introduction

THIS ARTICLE GREW out of some of the knowledge graph (KG) research and development topics discussed as part of the Ontology Summit 2019. Knowledge graphs essentially describe real-world entities (classes and instances) and their interrelations using a graph model. While the phrase “knowledge graph” can be found in the literature going back as far as at least 1972 (Schneider. 1973) and the idea of KGs can be found in the early days of artificial intelligence (AI) systems and semantic nets; the concepts evoked by this phrase have turned into a real activity area of work. One can hope there will be a general acceptance of an effective definition of "knowledge graph," since KGs are now a major focus of applied research to develop convenient, queryable information artifacts made from multiple sources of data and information (Noy *et al*., 2019). However, there remain many challenges in the KG design, assembly and implementation processes needed to enable a KG to reengineer information from a typically raw, messy, and disconnected state. As a simple totality the original, opportunistic sources of data for a KG, such as on the Web are incomplete, and often hard to query, analyze, and visualize. As they are gathered these must be cleaned and harmonized to a more refined, organized, and linked product that is easier to visualize, query, and analyze. Challenges as part of constructing a KG information artifact exist at every pipeline step of the data lifecycle process. These include many recursive steps within construction to refine and validate the information product. As an information project these are mixed in with external challenges such as scoping of information to use as well as sociotechnical challenges.

In this article I briefly overview these challenges that KGs need to address operationally at significant phases of work, as well as some overall problems that transcend stages. A big picture view of KG development is to see it as similar to system development, running from design, through development and testing to operational deployment. KGs have their own specifics at each stage. Moreover, drilling into phases of work, there are many intersecting steps during development of a knowledge base to support a KG. One may think of some refinement steps as reactions to failed tests of quality that cycle back to some earlier work on knowledge structuring and integration.

For the sake of exposition, phases of work such as entity identification or feature alignment are discussed somewhat separately as part of a typical sequence. Thus, extraction precedes entity identification that leads to entity refinement, and later entity integration and graph completions. Each reflects its own processes and employs distinguishable methods, although they interact and may support one another recursively. It is important to add that due to space limitations, the challenges described in the following sections are not a comprehensive listing of challenges of building and operating a KG system. Rather, they represent issues that were encountered and inspired by discussion topics as part of the 2020 Ontology Summit.

The Mix of External and Internal Challenges along the KG Lifecycle

Developmental steps for a KG start with the external problem of initial data scoping. As part of this domain and data experts identify a core set of the best available data, information and semantic resource sources to be assembled and integrated. This typically reveals a vast space of possibilities to consider and to pare down to a scoped space. Within an established scope, KG development moves from exploring this data space to knowledge and data acquisition. The hope is to acquire both a schema to organize the graph and quality data in order to populate it. But there may be conflicting organizational schemas and/or too little structured data to seed a well-structured graph to start with, and work may proceed bottom up driven by data.

Scoping and Initial Models

Preceding the building of a KG are issues around the scope (and source) of the knowledge needed by the system. To illustrate problems I take here a wide view and put an emphasis on enterprise level KGs (EKG) systems which have larger data acquisition issues than smaller, standalone applications that may have a narrow domain focus.

Scoping analysis identifies the network of potentially relatable entities, such as available from datasets, relational databases, spreadsheets, XML, JSON, Web APIs. Linked Data etc. Together these contain a large amount of structured data together with unstructured raw and poorly documented data that can be leveraged to build and augment a knowledge graph (Blomqvist *et al*., 2010). These should provide a base of knowledge to satisfy competency questions and related queries, which are used as part of guiding methodologies such as eXtreme Design (XD) for building a KG. Because data in the KG's target space is heterogeneous, the scope of EKG coverage typically comes initially without a common model. Moreover, while there are many ontologies now available, usually they do not cover the scope of an EKG. In some cases several overlapping ontologies and/or conceptual models may be available. However, merging and converging these comes with many issues, such as how to manage differing hierarchies or harmonizing definitions and axioms. More details on addressing some of these issues is discussed further in another article in this special edition (Berg-Cross, 2021). It is simple to say in short that developing or adopting a unified model within a projected KG scope is a challenge and is often deferred for a while. Instead, to get underway a loose or lightweight model is often assembled bottom up from various sources or from simple schemas without too deep a consideration of semantic issues. Richer semantic models may be developed along the way or at a later stage of KG maintenance.

Structural issues with Populating and Validating the Knowledge in a KG

Given identification of a scope, efforts proceed to knowledge graph population and curation. Auer *et al.*'s (2018) best practice for population is to use an infrastructure, such as search and extraction tools to access four complementary sources of data/information:

1. First, the infrastructure leverages existing metadata, data, taxonomies, ontologies, and information models. A standard approach for populating a KG, as mentioned before, is to use data stored on the web. Some of that may be unstructured and poorly documented. Not all the data types and relationships will be obvious or correct. Similar concepts even if documented may cover different instances or decompose into different subtypes. Population may fall back on the intuitive semantics latent for human understanding in data labels. As a result, *ad hoc* efforts are often used to explore entity neighborhoods to find candidates for population. These may be as simple as comparing entities and values (Dong, 2020).
2. Second, an infrastructure then provides services, often with graphical assistance that enable direct contributions from scientists who describe their research, supported by intelligent interfaces and automatically generated suggestions.
3. Next it implements some degree of automated methods for information extraction, cleaning and linking.
4. Finally, it supports curation and quality assurance by stakeholders and other interested parties - domain experts, librarians and information scientists.

Whatever level of tooled infrastructure projects have, the acquisition starts with extraction (Dong, 2020). A common wisdom is that the hardest part about building a new KG is everything that happens as data is acquired and before the end product of queries are implemented. However, the fact is that KGs need to extract massive collections of interrelated facts and the underlying data needs to be cleaned. Some automation such as statistical techniques can be used to find anomalies such as misuse of datatype properties in the data (Pujara, Eriq, and Getoor. 2017). Even an initial population of KGs may include many data instances, so they quickly become large enough to hamper the efficiency of the tooled infrastructure for cleaning and checking mentioned above. This also challenges the data quality inspection and testing done by people.

Scoping should have identifying core data, but the next steps dealing with the reality of mapping relationships and understanding key data constraints. Again, a range of automation can help, including natural language processing (NLP) and text or data mining. Extraction from DBs and online linked data is the more familiar part. The structured extraction is similar to querying, while text processing is similar to NLP, but is enhanced by pre-processing. In a pre-processing step, the input, say a collection of online pages with text, can be classified by a template and clustered by another template. For extraction of information from images processing is more like computer vision analysis where one first extracts the numerical features from the text or images and then gives those visual features as input to a machine learning (ML) model. Both types of extraction can provide a candidate base of data extracted facts to work with.

The next phase of work transforms these candidate facts into a large, useful knowledge graph. This is itself formidable due to structural issues. As mentioned, KGs are heavily populated from unstructured data. This is the problem of "noisy data". The state of the art to handle this includes using statistical techniques to enhance data fit, aligning features and entities properly. Besides unstructured sources, others are semi-structured. They are not formal and, on the whole, may not be well-structured. This means, for example, that source extractions, including those using automation, which have some structure like linked data, may reflect overly simplified or inaccurate information. Online semi-structured information such as Wikipedia and the related DBpedia represents such sources that are low hanging targets. Another, crowd-sourced base for building a KG is Wikidata, which has more than 25k active users and employs 329 bots. It is a tempting source since it contains more than a billion statements about 92 million entities (Arnaout, 2021). Parts of these sources, like Wikipedia infobox tables, are often used for early population. However, several major challenges have been noted with this source, including:

1. Deterministic extraction patterns used in DBpedia (and other sources) are dynamic, hence vulnerable to template changes;
2. While links may provide valuable information about a relationship link, the labels may be ad hoc. Too much reliance on labels such as

Wikipedia links can lead to entity disambiguation problems (handling the ambiguity of natural language labels is discussed below);

3. Naive heuristic based extraction of unlinkable entities yields low precision, which hurts both accuracy and completeness of the final KB (Peng et al., 2019).

Both unstructured and semi-structured extractions have to be made to fit together. Moreover, they have to harmonize with the semantic implications each other implies and any formalized knowledge used. Most of the work on automatically mapping structured and semi-structured sources to ontologies focuses on semantic labeling (Qiu *et al*., 2018). But there are challenges. Automated systems such as the Never Ending Language Learner or NELL (Mitchell *et al*., 2018) may extract a fact from Wikipedia with great confidence. However, human and linguistic analysis, such as case analysis, show they can be wrong. Roles an entity plays can be difficult to determine without extensive context. A superficial process sees a role as an "actor" when in reality the entity is a "coach" (Padia, Ankur, 2017). KG developers might make use of linguistic analyses to help overcome the ambiguities of the use of natural language terms labeling data to arrive at formal distinctions for intended interpretations.

Entity/Feature Alignments Entity Refinements, and Naming Resolution

Feature extraction and entity alignment take information from multiple schema types (*e.g*., CSV, data tables) to some form of a common graph-ready schema. This provides organization but may be well short of formal ontology semantics. In the large volume of data spaces, feature alignment and managing entity identities from heterogeneous data sources pose several obstacles even if an underlying model has been crafted. Entity names/labels are often not reliable. Entity resolution, the merging of records that refer to the same entity, is thus a key problem. Do, for example, the labels "born" (say "2001") and "date of birth" (say 5/4/2001) mean the same thing (Pham *et al*., 2016), and do they align with one entity in the model? Linking entity as part of alignment is enabled if a governing schema, hierarchy or conceptual model, such as mentioned above, has been developed. Mapping data to a shared schema or ontology is considered a key step in KG development (Auer *et al*., 2018; Ehrlinger and Wöß. 2016), since aligning with a domain ontology brings the additional benefit of formal semantics, which in turn can help with later alignments.

While building a KG, mature processes are needed to find entity types in unstructured data (Taheriyan, Knoblock *et al.*, 2016). Important information to identify entities and features may also be found in images on the Web. Images are an important and easy source of interpretable, contextual knowledge for humans. However, until recently, automated feature extraction from images was hard. Machine learning techniques using deep neural nets have made a difference, but it is still challenging to align entities and extract feature information from images in a way that is meaningful to humans. It is a sobering fact that feature type ontologies reflecting people's understanding may have 1000 types to choose from (Yan *et al.*, 2021). Moreover, these are often hidden in headings that convey key relations, attributes, or qualification such as the valid dates of facts.

Refinements in ideas such as capturing asserted facts about identity over time from data is a special type of refinement and obviously also challenging. For a KG, alternate sources may claim different, relevant periods and there may be gaps in data that need to be filled wisely. Work with linked data has shown that by analyzing the co-occurrence of topics and entity types, new types can be assigned to entities based on the topics/types found for those entities (Sleeman, Finin, and Anupam, 2015).

Resolving entity identity for moderately sized data sets manually is a bottleneck. By one report, it can take up to 6 months (Hertling and Paulheim, 2018). Some type of entity and name resolution is especially important where alternate textual formulations are used. Again, the size and scale of possible alternative in an information space makes resolutions a challenge. As part of population from DBs, for example, there are too many tables with impossible/incorrect/incompatible labels (who is Doctor "anonymous"?). Moreover, it is hard to join the tables since data was originally modeled for particular applications and not for integration as needed in a KG. Machine learning (ML) automation for names entity resolution (NER) is one place to look for help (Yadavan and Bethard. 2019).

While there remain limits to automated feature and entity resolution, it is also true that extraction using more mentalistic identification and labeling of features can be misleading due to human biases. This is especially likely if extraction is done by data or computer scientists lacking domain experience. Best practice advice by Knoblock (2018) is that it helps to start with sources using semantic labeling that annotates data fields with ontology classes and/or properties. However, a precise mapping that fully recovers the intended entity meaning from data needs to describe the semantic relations between the data fields too. This provides context and shows how good work on one process step to make subsequent steps such as graph integration easier.

There remain entity resolution challenges that come from noisy data of any kind, which raises the question, "Just what is meant?" Knowledge refinement addresses some types of noise like missing knowledge, redundancies or just plain erroneous knowledge. Statistical and ML techniques can and have been used, but we do not yet fully understand the range of possible errors that could occur in extracted facts over various domains (Zou *et al*., 2020). Resolving entity identity remains for now a mixed and balanced process of some automation and some manual, holistic clean up activity.

One should note that there is a big role of ML driven embedding methods in extractions needed to develop KGs. Knowledge graph embedding refers to the embedding of components of a KG including entities and relations into continuous vector spaces. This is used to simplify the refinement of a KG while preserving its inherent structure. Embedding is used by a variety of downstream tasks such as KG completion and relation extraction, and hence has gained some attention as a useful practice. A representative approach embeds KBs into latent spaces and makes inferences by learning and operating on latent representations. Such embedding models, however, do not make use of any rules during inference and hence one suspects have limited accuracy (Lin *et al*., 2015; Wang, Wang, and Guo, 2015; Wang *et al*., 2017).

Graph Integration and Graph Completion

Graph integration relies on handling entity and feature alignment issues, and some early opportunistic integration may happen as part of alignment. If a general schema or ontology has been used integration may take place a bit more routinely. However, early integrations are often only partial and a final phase of integration with validation testing is needed. Without a broad and deep graph integration, we can ask, "Can KG efforts be integrated or are we building silos at a different level?"

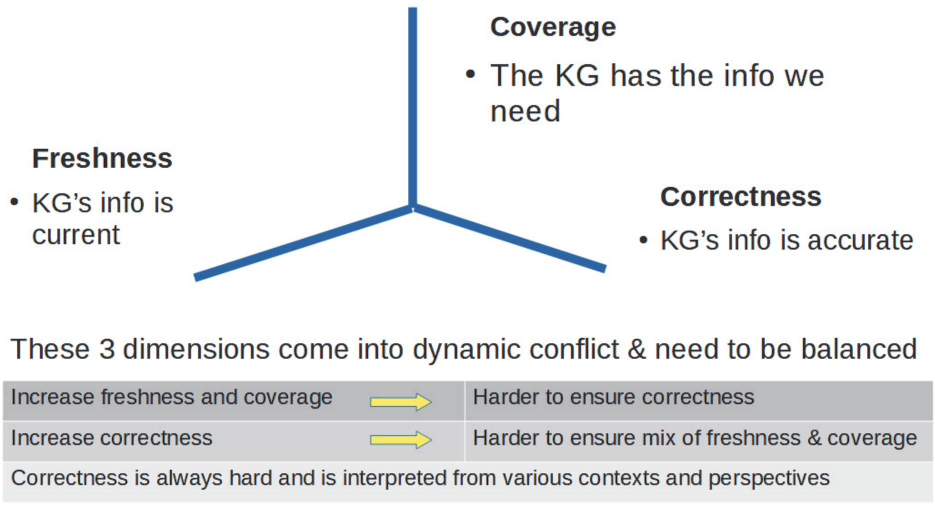
Beyond the immediate problem driven by sources, KG integration difficulties come in several forms. Some are again structural, and some are due to gaps to fill:

- Integration of data coming from a variety of source locations presents challenges beyond the entity naming issues that were previously mentioned. The sources may use different data organizational schemes. Named entities and extracted relations may represent mined information that needs to be further integrated with existing structured data (*e.g.,* via Entity Linking techniques) in order to yield relatively complete entity descriptions;
- One way to view the challenge of graph integration and well as eventual completion is to consider some integration characteristics of the main online sources for population. Wikipedia provides information for a KG early on, but has what is called knowledge incompleteness. For example, in 2014 only 46% of person entities in Wikidata have birthplaces available, according to Vrandeci and Krötzsch (2014). All of this gap filling along with error correction is needed before KG construction is complete.
- The integration be targeted to more than one application, which require different (heterogeneous) data organizational schemes. This is known as KG to App integration. As mentioned, alignment with a community schema or ontology supports this later step of wider integration.

The integration reality for KGs is the need to handle complex relationships as they move from entities to entity integrations. To reflect some of their domain knowledge, KGs can include many complicated relations to handle things as roles and situations that lay things together. These can reflect relations used to capture specific contexts as well as level(s) of abstraction involved in the conceptualizing hierarchies (Kim *et al*., 2015). Importing data for new purposes differing from their original siloed means harmonizing and meaningful mapping entities but also relations between entities. Depending on the degrees of differences in conceptualization, this can make KG integration step a demanding task.

Knowledge graph completion refers to determining if there is a relation between two graph entities and, if so, specifying the type of the relation. One may think of this as completing a triple using 2 entities A and B and finding a relation R to form the triple. Knowledge graph completion may use ML embedding techniques to predict relations between entities under supervision of the existing KG.

One known problem reflecting a KG comprehensiveness, if not completeness, is that many data sources used for entity extraction (per previous examples) can create representativeness problems. As a necessity, large-scale KGs also have to make some type of trade-off between knowledge completeness and correctness. As shown in Figure 1 adapted from Gao (2018) there are really three key KG dimensions in conflict and requiring balance during development. These start with the consideration of correctness and coverage, but also the idea of completeness of a KG along with the timely freshness and veracity of the information. By KG correctness, we don't mean the graph always knows the "right" value for an attribute to answer a query. Rather, it means that the knowledge in a KG is always able or competent to explain why a certain assertion was asserted. The test of correctness is that an assertion should reflect a consensus and make sense to a domain expert. Data provenance about sources captured during data acquisition and documentation of integration trade-offs also provides some explanatory basis for A KG's correctness. Following initial graph population final, tuned construction often includes new links and confidences about facts and relations which advance completeness. This also helps with big data performance, such as the ability of a KG to handle fast data in real-time. However, working, conceptually elegant designs for the first portion of data may not scale up as more instances and types of data are added to complete a KG. Later, the same issues are faced as part of maintenance and updates. As more data is acquired, different vocabularies are introduced and different patterns may encode the same attribute. To mitigate this challenge, as discussed in another article in this special edition (berg-Cross, 2021), graph construction should be viewed as an incremental process with a final assembly that includes a check of semantic relations, if possible, from a guiding ontology.



Coverage

- The KG has the info we need

Freshness

- KG’s info is current

Correctness

- KG’s info is accurate

These 3 dimensions come into dynamic conflict & need to be balanced

Increase freshness and coverage	⇨	Harder to ensure correctness
Increase correctness	⇨	Harder to ensure mix of freshness & coverage
Correctness is always hard and is interpreted from various contexts and perspectives		

Figure 1: Three key KG dimensions to balance - adapted from (Gao, 2018)

Knowledge Representation and Query Languages

As previously noted, data populating a KG may be modeled as RDF triples. They are readily available, and the argument for them is that they enabled easy data management and provides a simple way to fold in some semantics for existing data. KGs tend to be less formal than the best ontologies and follow the Linked Data component of the Semantic Web approach, using RDF/RDFS to express simple factual information. However, RDF has obvious limited expressiveness compared to natural language or ontologies. Indeed, as Pat Hayes pointed out, formalisms like RDF lack expressivity. Semantic nets of the 1970s were, almost unilaterally, much more expressive than knowledge graphs or RDF, or any of the other ‘graph’-like modern notations. Semantic nets typically had ways of encoding quantifier scopes, disjunction, negation and sometimes such things as modal operators.

Since RDF/RDFS has limited semantic relations, over time KGs based on them encounter the limits to reasoning. This is a problem documented years ago (Sowa, 2011). The long-term implications are clear: the continued use of limited representation require semantic resources like ontologies to boost knowledge expressiveness. A variety of incremental approach to achieve this is discussed in Berg-Cross (2021).

Graph Construction, Performance, Queries, Scalability and Maintenance

Big data comes with the dimensions of volume and velocity. The challenge is to manage changing knowledge due to the fast incremental updates that are feeding large-scale KGs such as KGs that are being scaled up to handle such problems as epidemic and hospital data. Any effective entity-linked KG structure will grow based on its ever-changing expanse of related data. For example, even standard organizational knowledge represented in a KG may merge or split. New scientific discoveries may break an existing concept like coronavirus into subtypes (Paulheim, 2017). Since size makes it impossible to validate and verify KG updates manually, like initial KG population, it is tempting to automate the verification process. Automation maybe especially effective if data structures are not changing, but instance data is being added. Unsupervised and semi-supervised knowledge extraction from unstructured data in relevant domains is one approach. However, these may again be open to many different interpretations of domain knowledge that have to be resolved. Advances needed for some degree of automated support include:

- better domain knowledge representation and reasoning,
- probabilistic models for adjusting graphical structures and
- natural language inferences that can be used to construct an automatic or semi-automatic system for consistency checking and fact verification.

Selection of a graph base and a system for deployment of a KG is the proverbial last mile. It often receives like discussion, but is actually a diverse problem area, including the obvious effect it can have on performance or on associated tools such as graphical interfaces. Real-time KG operations is a major factor in selection. Graph databases are not known for their speed or scalability. In fact, they are generally speaking the smallest and slowest of data model types and rich semantics are not yet easily adapted for dynamic and responsive application and KG environments (Wei, 2018). Scalability is also a KG issue at all phases of the KG life, and notable for performance (but also maintenance). All the graph models noted by (Rajangam, and Annamalai, 2016) share some common limitations. For large knowledge bases, the graphs become too large to perform required operations within convenient time. Moreover, changes in existing knowledge can increase the overhead cost of maintaining the graph's nodes and edges. Scale issues manifests themselves indirectly by affecting other operations, such as managing fast incremental updates to large-scale KGs. Traditionally, to help performance the operative part of the graph stays in the RAM, but multiple threads can access it; and, as with most KG challenges, there are trade-offs to consider. When performing real-time operations, it is necessary to consider the time of execution, but also to respect the quality and precision of execution.

Associated with representation issues, there are graph DB performance concerns with modeling certain data types. Time series, for example, are not well expressed in KGs that use simple RDF. This makes for some *ad hoc* structures with processing issues, but some workarounds and new representations for temporal information have been proposed (Leblay and Chekol, 2018).

Besides sheer query performance, most, if not all KG systems, face the challenge of managing the graphs at scale over time. This requires a proper infrastructure. Obviously, a KG infrastructure must include a scalable graph-storage backend to store information and expose a comprehensive API for interacting with the KG.

There are several graph database languages on the market that address both these query performance and graph maintenance. These include Neo4j's Cypher, Google Cayley, TIBCO, Apache TinkerPop Gremlin, Amazon's Neptune and TigerGraph's GSQL *etc*. Selection involves not only performance, but also which query language and its expressiveness a team is comfortable with. New query languages for KGs (like Cypher) exist, but standardization remains a current area of concern for industry stakeholders.

When considered a graph database, other evaluation factors include: ability to deal with all of the previous problem areas mentioned, such as:

- Schema and modeling flexibility and independence
    - This includes graph management capability to design and execute complex algorithms beyond simple queries to exert efficient and granular control of both the graph query and the graph model elements, *i.e*., editing of vertex and edge instances as needed.

- Ability to import and leverage complex and semantic sources:
  - ontologies, taxonomies, vocabularies
- Linking such as mapping datasets, vocabularies, *etc*. to the KG structure
- Efficient traversal of graph nodes such using parallel semantic processing (Beneventano and Vincini. 2019)
- Privacy and security
  - Not all the KGs use security mechanisms for access. So it is necessary to classify the KG and use tools for analysis before processing.
- Exploration of data via complex GUIs

External Challenges

Besides these internal, construction, performance and maintenance hurdles, there are external requirements that can frustrate KG success. External challenges identified by Sheth *et al*. (2019) include capturing context and domain-specific knowledge issues. Context exists in KG neighborhood structures, but may not reflect human understanding and reasoning about an entity and its context. The pre-conscious background knowledge and related reasoning engaged in as part of human understanding seems very different from what exists in current KGs. Some context is captured in populating KGs and ranges from spatial and temporal information and related reasoning along with provenance. However, context for human style reasoning, learning and commonsense understanding are largely unaddressed in current work. Instead, the reasoning that is typically available as a part of a KG in simple logical inference, graph node-wise reasoning, such as search, link predication, entity prediction, or subgraph matching (Liu *et al*., 2020). The difficulty of adding commonsense reasoning in particular employed over a large base of commonsense knowledge captured in a KG remains.

Related questions include how to handle implicit relations, strength of (causal) relations, and exclusiveness (Popping, 2003). This is a recognized gap that some hope to start to fill with intelligent text analysis and by the crawling and analyzing of relevant sites and social media in real-time as a better source of commonly understood and used knowledge (Ilievski, Szekely, and Zhang, 2021).

The Internet of Things (IoT), like healthcare, represents an example of an external challenge of interest that Sheth *et al*. (2019) feature. It is a complex domain with many interacting specific subdomain parts that may involve commonsense reasoning and that promise big potential for successful applications. However, an IoT KG comes with specific knowledge issues, including how to automate the building of a large base of domain knowledge and cross disciplinary schemas from text to support a range of applications. A start on this has been made by Noura *et al*. (2019) to test how well existing ontologies in subdomains match up to concepts extracted from IoT text.

To these examples we might add the continuing challenge of adequate visualization as part of user interfaces. Visualization methods remain the main means to support KG usability, analysis if completeness and clarity and to provide both big picture and drill down detail understanding. Early efforts and visualization tools (graph-based or template-based visualization) mainly reflected a self-limited ability to expose the syntactic structure of KGs rather than their conceptual semantics (Desimoni & Po, 2020). They lacked the flexibility to easily visualized diverse parts of a graph or allow users to specify information. Since multiple semantic resources, such as ontologies and ODPs may be used to craft a KG, their visualization is important too. Interfaces that allowed source comparisons of multiple resources were not available until very recently and are still in early stages of development (Asprino, Carriero, and Presutti, 2021). Finally, visualization serves as an explanatory tool. Users often need an interface with a modest explanatory ability to describe the relative important of attributes as part of query relaxation or refinement when displaying results.

Discussion of Opportunities and Future Directions

In this paper, I have reviewed some of the challenges in KG development and use. Despite the challenges, KGs have started to have a significant effect on data and knowledge management in particular areas. This is likely to grow into a more general impact. To achieve this, there remain a large suite of problems with activities that are internal to the KG lifecycle. This includes phases of works from scoping through population, alignment and refinement to completion of graphs. There are potential trade-offs within and between each and every process along the way. Within the KG lifecycle, semantic and data technologies are of obvious value and play a role. A variety of different methods, ranging from NLP and ML techniques, are increasingly used, but they yield approximate products and cannot yet adequately automate solutions to commonsense problems like resolving entity identity over time. There remains a need to address development of common tools that can interoperate across the KG lifecycle. Supervised ML is one area of notable research and is increasingly important in extracting text and images. However, a concern is that there is not enough training data to support robust ML and deep learning systems, especially in complex and promising areas like IoT. Approaches to overcome this can adopt fully unsupervised ML approaches (*e.g*., clustering with vector representations) or semi-supervised techniques such as distant supervision with existing knowledge, multi-instance learning, active learning, *etc*. (Casolla *et al*., 2019). While automated help is advancing, real reasoning limitations have been noted. For example, we do not yet know how to integrate logical views with statistical ones. Statistical methods as reflected in deep neural networks do not readily provide information about and for the process of "reasoning" or "deduction". This generates problems for applications, including KGs where explanation and dialog are needed for users and developers. And one can hope that KG developers will understand the need for a well thought out schema and how ontologies, or at least ontological analysis, can aid in KG semantic improvement. All of these challenges within the vision of manageable KG systems and infrastructure architecture handling need to be addressed. It seems likely that there will transitional systems to incorporate KG systems and supporting infrastructure into more traditional and front-line dynamic information systems,

As of yet, scalability issues have not been systematically researched for most aspects of the KG lifecycle, as well as operational performance. Mining and refining associated neighborhoods and paths in large graphs, for example, is only starting to be addressed.

Conclusions

In conclusion while challenges remain, there are now active technical research areas to support the rapidly expanding space of KGs in order to align and semantically unify richly interconnected heterogeneous data. It is encouraging that, despite the difficulty of population efforts using multiple sources, we are becoming better at building KGs with less noise at each phase of work. Among the remaining challenges are those of ontology merging, developing an adequate base for ML, agreeing on an adequate approach to situational and contextual understanding, and understanding how to use deep learning in dynamic situations. It is especially important to support the need to keep humans in the loop with the variety of automation being developed, such as ML-generated models. Moreover, there remains the need for a common, enhanced ontology engineering practice addressing the of structuring the semantics of KGs.

References

Arnaout, Hiba *et al*. "Negative Knowledge for Open-world Wikidata." *Companion Proceedings of the Web Conference 2021*. 2021.

Asprino, Luigi, Valentina Anita Carriero, and Valentina Presutti. "Extraction of common conceptual components from multiple ontologies." *arXiv preprint arXiv:2106.12831* (2021).

Auer, Sören *et al*. "Towards a knowledge graph for science." Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics. ACM, 2018.

Beneventano, Domenico, and Maurizio Vincini. "Foreword to the Special Issue:“Semantics for Big Data Integration”." (2019): 68.

Berg-Cross, Gary, Issues in Incrementally Adding Better Semantics to KGs. 2021

Casolla, Giampaolo *et al*. "Exploring unsupervised learning techniques for the Internet of Things." *IEEE Transactions on Industrial Informatics* 16.4 (2019): 2621-2628.

Blomqvist, Eva *et al*. “Experimenting with eXtreme Design”. In: Proc. of EKAW 2010. (Lisbon, Portugal). Vol. 6317. Springer, 2010, pp. 120–134.

Desimoni, F., Po, L.: Empirical evaluation of linked data visualization tools. Future Generation Computer Systems 112, 258–282 (2020)

Dong, L. "Knowledge graph and machine learning: A natural synergy, presentation at Stanford seminar on KGs." (2020).

Ehrlinger, Lisa, and Wolfram Wöß. "Towards a Definition of Knowledge Graphs." *SEMANTiCS (Posters, Demos, SuCCESS)* 48.1-4 (2016): 2.

Hertling, Sven, and Heiko Paulheim. "Dbkwik: A consolidated knowledge graph from thousands of wikis." 2018 IEEE International Conference on Big Knowledge (ICBK). IEEE, 2018.

Ilievski, Filip, Pedro Szekely, and Bin Zhang. "Cskg: The commonsense knowledge graph." *European Semantic Web Conference*. Springer, Cham, 2021.

Kim, Dokyoon *et al*. "Knowledge boosting: a graph-based integration approach with multi-omics data and genomic knowledge for cancer clinical outcome prediction." *Journal of the American Medical Informatics Association* 22.1 (2015): 109-120.

Knoblock, Craig. From Artwork to Cyber Attacks: Lessons Learned in Building Knowledge Graphs using Semantic Web Technologies, U.S. Semantic Technologies Symposium March 1, 2018.

Leblay, Julien, and Melisachew Wudage Chekol. "Deriving validity time in knowledge graph." *Companion Proceedings of the The Web Conference 2018*. 2018.

Lin, Yankai *et al*. "Learning entity and relation embeddings for knowledge graph completion." *Twenty-ninth AAAI conference on artificial intelligence*. 2015.

Liu L, Du B, Ji H, Tong H. KompaRe: A Knowledge Graph Comparative Reasoning System. arXiv preprint arXiv:2011.03189. 2020 Nov 6.

Mitchell, Tom *et al*. "Never-ending learning." *Communications of the ACM* 61.5 (2018): 103-115.

Noura, Mahda *et al*. "Automatic Knowledge Extraction to Build Semantic Web of Things Applications." *IEEE Internet Things J.* 6.5 (2019): 8447-8454.

Peng, Boya *et al*. "Improving Knowledge Base Construction from Robust Infobox Extraction." *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*. 2019.

Pham, Minh *et al*. "Semantic labeling: a domain-independent approach." International Semantic Web Conference. Springer, Cham, 2016.

Popping, Roel. "Knowledge graphs and network text analysis." *Social Science Information* 42.1 (2003): 91-106.

Pujara, Jay, Eriq Augustine, and Lise Getoor. "Sparsity and noise: Where knowledge graph embeddings fall short." *Proceedings of the 2017 conference on empirical methods in natural language processing*. 2017.

Qiu, Jing *et al*. "Automatic non-taxonomic relation extraction from big data in smart city." *IEEE Access* 6 (2018): 74854-74864.

Sleeman, Jennifer, Tim Finin, and Anupam Joshi. "Topic modeling for rdf graphs." *3rd International Workshop on Linked Data for Information Extraction, 14th International Semantic Web Conference*. Vol. 1267. 2015.

Edward W. Schneider. 1973. Course Modularization Applied: The Interface System and Its Implications For Sequence Control and Data Analysis. In Association for the Development of Instructional Systems (ADIS), Chicago, Illinois, April 1972.

Gao, Yuqing *et al*. "Building a large-scale, accurate and fresh knowledge graph." *SigKDD*. 2018.

Padia, Ankur. "Cleaning noisy knowledge graphs." *Proceedings of the Doctoral Consortium at the 16th International Semantic Web Conference*. Vol. 1962. 2017.

Sheth, Amit, Swati Padhee, and Amelie Gyrard. "Knowledge graphs and knowledge networks: The story in brief." *IEEE Internet Computing* 23.4 (2019): 67-75.

Sleeman, Jennifer, Tim Finin, and Anupam Joshi. "Topic modeling for rdf graphs." *3rd International Workshop on Linked Data for Information Extraction, 14th International Semantic Web Conference*. Vol. 1267. 2015.

Sowa, John F. "Future directions for semantic systems." *Intelligence-based systems engineering*. Springer, Berlin, Heidelberg, 2011. 23-47.

M. Taheriyan, C. A. Knoblock *et al*., “Leveraging Linked Data to Discover Semantic Relations Within Data Sources,” in ISWC, 2016.

Vrandečić, Denny, and Markus Krötzsch. "Wikidata: a free collaborative knowledgebase." *Communications of the ACM* 57.10 (2014): 78-85.

Wang, Quan, Bin Wang, and Li Guo. "Knowledge base completion using embeddings and rules." *Twenty-Fourth International Joint Conference on Artificial Intelligence*. 2015.

Wang, Quan *et al*. "Knowledge graph embedding: A survey of approaches and applications." *IEEE Transactions on Knowledge and Data Engineering* 29.12 (2017): 2724-2743.

Yadav, Vikas, and Steven Bethard. "A survey on recent advances in named entity recognition from deep learning models." *arXiv preprint arXiv:1910.11470* (2019).

Yan, Bo *et al*. "Harnessing Heterogeneous Big Geospatial Data." *Handbook of Big Geospatial Data* (2021): 459.

Zhu, Q., Wei, H., Sisman, B., Zheng, D., Faloutsos, C., Dong, X. & Han, J. (2020) Collective multi-type entity alignment between knowledge graphs. In *WebConf 2020*.