Journal of Systems Engineering and Electronics
Vol. 35, No. 3, June 2024, pp.679 – 688

How to implement a knowledge graph completeness assessment with the guidance of user requirements

ZHANG Ying* and XIAO Gang

Institute of Systems Engineering, Academy of Military Sciences, Beijing 100107, China

Abstract: In the context of big data, many large-scale knowledge graphs have emerged to effectively organize the explosive growth of web data on the Internet. To select suitable knowledge graphs for use from many knowledge graphs, quality assessment is particularly important. As an important thing of quality assessment, completeness assessment generally refers to the ratio of the current data volume to the total data volume. When evaluating the completeness of a knowledge graph, it is often necessary to refine the completeness dimension by setting different completeness metrics to produce more complete and understandable evaluation results for the knowledge graph. However, lack of awareness of requirements is the most problematic quality issue. In the actual evaluation process, the existing completeness metrics need to consider the actual application. Therefore, to accurately recommend suitable knowledge graphs to many users, it is particularly important to develop relevant measurement metrics and formulate measurement schemes for completeness. In this paper, we will first clarify the concept of completeness, establish each metric of completeness, and finally design a measurement proposal for the completeness of knowledge graphs.

Keywords: knowledge graph completeness assessment, relative completeness, user requirement, quality management.

DOI: 10.23919/JSEE.2024.000046

1. Introduction

Knowledge graph quality assessment refers to the assessment of data and relationships in the knowledge graph, which needs to be based on the needs of specific application scenarios to determine the quality of its accuracy, completeness, consistency, and other dimension. It provides guidance and guarantee for construction, management, and application of knowledge graphs [1,2], and further promotes the development and application of the knowledge graph. In its actual assessment process, researchers usually set various assessment dimensions, indicators, metrics, and algorithms to assess the quality of the knowledge graph and propose corresponding improvement strategies and techniques to improve the quality and credibility of the knowledge graph. As an indispensable element in the quality assessment of knowledge graphs, completeness assessment needs to be conducted in the context of relevant tasks by considering the actual needs [3–7]. More specifically, whether a knowledge graph is complete or not needs to depend on the use case scenario, which means that it depends on the given knowledge graph and the user's current specific needs [3,4]. Among them, schema completeness, property completeness, and population completeness are three metrics commonly used to assess completeness [2,8]. Schema completeness is often evaluated by comparing the knowledge graph to be tested with the "gold standard" [4,9–15]. Property completeness is based on the determined schema and queries are used to determine whether the number of property values/tail entities satisfies a predefined ideal threshold [14,16,17]. Similarly, population completeness is similar to schema completeness and can be evaluated by comparing it with the "gold standard" [4,9,14]. It can also be evaluated by trying to determine the number of entities that should be included under a certain category using related algorithms [18–20].

Most methods for assessing completeness currently use the same paradigms applied to different types of knowledge graphs. However, there is a certain degree of difference between the intrinsic meaning of completeness and the existing assessment techniques, and the assessment process cannot consider the need for completeness to be assessed in the corresponding application context. As a result, the final evaluation results can hardly provide a better reference value for users [21,22]. In addition, existing assessment methods do not provide a clear method for

Manuscript received May 23, 2023.
*Corresponding author.
This work was supported by the National Key Laboratory for Complex Systems Simulation Foundation (6142006190301).

obtaining a baseline such as the “gold standard”, and most of them reuse the existing “gold standard” proposed by experts to conduct completeness assessment in the corresponding domains. As a result, completeness assessment methods that involve reliance on the “gold standard” often have difficulty in considering the actual needs of users and lack appropriate feedback on user needs. In addition, this type of approach lacks scalability to a certain extent and requires the design of corresponding “gold standards” based on the characteristics of different domains. Therefore, it is difficult and time-consuming to replicate such an evaluation model in many other domains [21]. Based on the above considerations, this study is the first to incorporate user requirements into the completeness assessment technique. The assessment of completeness enables users to understand the completeness level of the knowledge graph and helps them select a knowledge graph that is more compatible with the current task. In addition, data providers can rely on the assessment results to know the lack of knowledge in the knowledge graph, and thus mine the relevant information to improve the completeness level of the knowledge graph in a targeted manner [14]. Further, the results can be used to design corresponding knowledge graph construction techniques and enhance dynamic updating capabilities, so that they can meet the individual demands of a wider range of users in the future. In addition, we provide a scalable approach to obtaining the “gold standard” for all domains, that is, mining the actual usage requirements of users as an important paradigm for completeness assessment. Also, by mining latent patterns, we can flexibly predict future latent needs and provide an informed reference for the direction of knowledge graph improvement, thus enhancing the applicability of the knowledge graph.

In addition, the proposed completeness assessment method can be used to assess the completeness of the current knowledge graph based on existing user search records and produce relative completeness assessment results that are specific to different usage needs and contexts at different periods. This enables potential users to understand the completeness and lack of knowledge of the current knowledge graph in a specific context and provides a basis for decision-making when choosing a knowledge graph in the future.

In this paper, we provide a more practical completeness assessment proposal, which redefines the concept of completeness and the corresponding completeness measurement metrics based on the user’s requirements. Also, concepts are proposed to enrich the new proposed completeness metric system for a more accurate completeness assessment. The rest of the paper is organized as follows. In Section 2, the newly proposed completeness metrics are introduced. In Section 3, the specific assessment proposal for the completeness dimensions is given. In Section 4, completeness assessment experiments are conducted with the existing dataset. Finally, the above completeness assessment techniques are summarized, and future research directions are given.

2. Relative completeness

When selecting knowledge graphs, users tend to prefer ones that are large in scale and volume, but such knowledge graphs do not necessarily mean that they contain the knowledge needed for the current task [23,24]. It is likely that this is due to problems such as redundancy and inconsistency of knowledge. Therefore, large size does not equal completeness of knowledge or fulfilment of user needs. In addition, when faced with the same volume of knowledge graphs, it is difficult for users to select knowledge graphs only by the number of triples, so a more specific completeness assessment of knowledge graphs is needed to determine which knowledge graph contains triples that better meet the actual needs of users. Meanwhile, there is a gap between the intrinsic meaning of completeness and the existing assessment methodology [22], and the specific calculation process does not effectively consider the user’s requirements [21]. Therefore, it is necessary to perform a user-needs-based completeness assessment for knowledge graphs. Furthermore, since knowledge is constantly expanding and updating, with the constant emergence of new research and discoveries, even good quality knowledge graphs require constant updating and quality enhancement to reflect the latest knowledge content [6,25–29]. Therefore, the updating of knowledge will inevitably lead to changes in the user’s requirements for the completeness of the knowledge graph, and the use of the “gold standard” for completeness assessment will result in a mismatch between the requirements and the assessment results. In addition, due to the limited nature of human knowledge, no knowledge graph can be called complete [30–32], even in a specific domain, the breadth and depth of knowledge is infinite. Therefore, the demand for the completeness of a knowledge graph needs to be placed within a reasonably limited range, which can be calculated, for example, by comparing the user’s needs with the knowledge contained in the knowledge graph, and calculating to what extent the current knowledge graph can meet the user’s actual needs.

Based on above, we have developed a more comprehensive definition called “relative completeness” that considers the user’s varying knowledge needs, building on the initial definition of completeness. Relative completeness reflects the extent to which a certain entity and its related information are needed by users in the knowledge graph by considering the frequency of access to different entities and their attributes as well as the requirement of the number of attribute values. For example, users access entities that are in line with current events or hot topics more frequently than other entities and demand more information about them. Therefore, from the user’s perspective, such frequently accessed entities will have a greater impact on the completeness of the knowledge graph. At the same time, even for the same knowledge graph, the user’s perception of the completeness of the knowledge graph will change to different degrees due to various factors such as the change of user’s needs, the frequency of knowledge update, and the change of time. To sum up, if the completeness of the knowledge graph is assessed simply by using the traditional way, there will be a certain lack of consideration of the nature of completeness itself, which will lead to the completeness result of the knowledge graph not being informative to the users.

2.1 Relative completeness concept

For completeness assessment based on user requirements, we propose the concept of relative completeness. Relative completeness refers to the ratio of the amount of knowledge contained in the knowledge graph to the amount of knowledge required by the user in a specific context of use at a specific time. This definition indicates that completeness needs to be measured concerning usage time and usage context. To illustrate “relative completeness” in more detail, the following three concepts are proposed to enrich it. First, the concept of entity prevalence is proposed to capture the impact of different entities on completeness. Second, for the same purpose, similar concepts are proposed for the number of properties and property values, namely, property prevalence and cardinality prevalence. Moreover, link completeness aims to examine whether there are valid links between equivalent entities. It is an enhancement of the consistency of the knowledge graph, so link completeness is not considered here. The final practical measurement of completeness will be specifically calculated for user usage requirements through the following three completeness metrics, that is, entity completeness, schema completeness, and property completeness.

Completeness as discussed in this paper refers to the proportion of the knowledge graph that contains the amount of knowledge required by the user. The amount of knowledge refers to the breadth of knowledge covered by the knowledge graph, which can be measured by the number of triples that can be retrieved. In different scenarios, the amount of knowledge contained in a knowledge graph can also be the number of its entities, the number of relationships, and the number of related properties.

Definition 1 Entity prevalence. In calculating entity completeness, different entities are assigned different importance weights based on the frequency of user access. The different weights are used to indicate the degree of influence of different entities on completeness.

Definition 2 Property prevalence. When calculating schema completeness, different properties are assigned different importance weights based on the frequency of access or preference of the user. The different weights are used to indicate the degree of influence of different properties on the completeness calculation.

Definition 3 Cardinality prevalence. The number of attribute values required by users is determined for different attributes when calculating property completeness. Here, the cardinality assumption is applied to determine the respective K-values for different properties, and the number of property values is considered to satisfy the completeness requirement of the user when and only when the number of property values is greater than or equal to the K-value.

2.2 Analysis of relative completeness measurement

In the actual measurement process, we need to capture the relationship between the user’s knowledge requirements and the amount of knowledge contained in the knowledge graph and produce practically meaningful completeness assessment results through actual analysis. For the convenience of representing the relationship between them, the following blue circles are used to represent the knowledge in the current knowledge graph, and the red boxes represent the knowledge required by users. As shown in Fig. 1, there are three cases from left to right, where the knowledge required by the user is fully available from the knowledge graph, the user’s demand for knowledge is much larger than the amount of knowledge contained in the knowledge graph, and the user can only obtain partial knowledge from the knowledge graph. In the following, the completeness of the knowledge graph is evaluated based on the above three cases, taking the search application based on the knowledge graph as an example. Among them, Completeness_g represents the relative completeness of the knowledge graph g in satisfying the user’s search demand in a certain period.

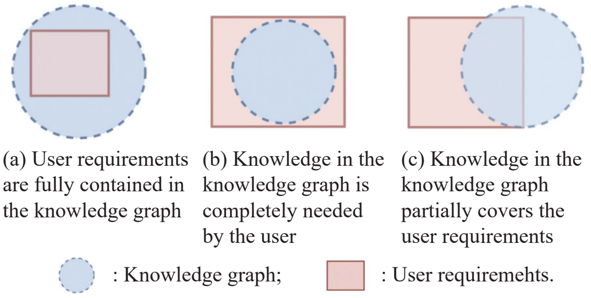


Fig. 1 Relationship between users' requirements for knowledge and knowledge graph

As shown in Fig. 1 (a), the knowledge contained in the knowledge graph is much more than what the user needs, so the completeness of the knowledge graph is 100% for the user, because all the required knowledge can be searched to get the corresponding results, at this time $\text{Completeness}_g = 100\%$. When the knowledge graph cannot meet the actual usage requirements of users, it can be subdivided into the following two specific cases. First, as shown in Fig. 1 (b), there is some users' required knowledge that cannot be searched for in the knowledge graph. For the user, the completeness of the current knowledge graph is the proportion of the knowledge in the knowledge graph to the knowledge required by the user's search. The result of the completeness assessment at this time can be obtained as follows:

$$\text{Completeness}_g = \frac{K_g}{K_{\text{ur}}} \tag{1}$$

where K_g represents the amount of knowledge contained in the knowledge graph, and K_{ur} represents the amount of knowledge required by the user.

Then, there is also a situation as shown in Fig. 1 (c), where part of the knowledge contained in the knowledge graph can satisfy part of the knowledge required by the user. Similarly, for the user, the completeness of the current knowledge graph is the ratio of the amount of knowledge that can be searched in the knowledge graph to the amount of knowledge required by the user's search. At this point, the complete assessment result can be obtained by

$$\text{Completeness}_g = \frac{K_g \cap K_{\text{ur}}}{K_{\text{ur}}}. \tag{2}$$

The amount of knowledge mentioned above is based on the entities obtained from user search records and their associated attributes and relationships. To better demonstrate the above assessment process, a simple example is used next to represent the completeness assessment of Fig. 1 (c) in a real situation.

Table 1 shows the user's search records in a certain period, and Fig. 2 shows the relationship between user's needs and the amount of knowledge in the knowledge graph, where h_1, h_3, and h_5 are the entities that can be searched. Based on (2), we can get the result of completeness assessment of the knowledge graph g based on the user's needs in this period. In the actual calculation process, the search weight is used to indicate the degree of user demand for entities, that is, entity popularity. In this case, the user's demand is regarded as the denominator "1", and the numerator is the search weight of the entities that can be searched in the knowledge graph. Against the above three cases, the final entity completeness assessment result can be calculated as 30% $\left(\text{Completeness}_g = \dfrac{0.1+0.05+0.15}{1} = 30\%\right)$. Based on the assessment thinking shown above, our method can complete the assessment of the completeness metrics while combining the user's requirements, so that it can get more specific and more informative results of the knowledge graph completeness assessment.

Table 1 User history search records

Searched entity	Number of searches	Hits(Yes/No)	Search weight
h_1	10	Yes	0.1
h_2	5	No	0.05
h_3	5	Yes	0.05
h_4	15	No	0.15
h_5	15	Yes	0.15
h_6	10	No	0.1
h_7	10	No	0.1
h_8	15	No	0.15
h_9	15	No	0.15

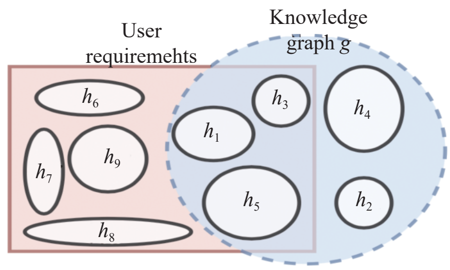


Fig. 2 Simple example of a user's knowledge graph-based search

2.3 Completeness metrics and calculation

As shown in Fig. 3, we calculate the entity completeness, schema completeness, and attribute completeness of the knowledge graph according to the above three cases in turn after obtaining the user's usage requirements. The three completeness metrics mentioned above and the calculation methods are described in detail in the following section.

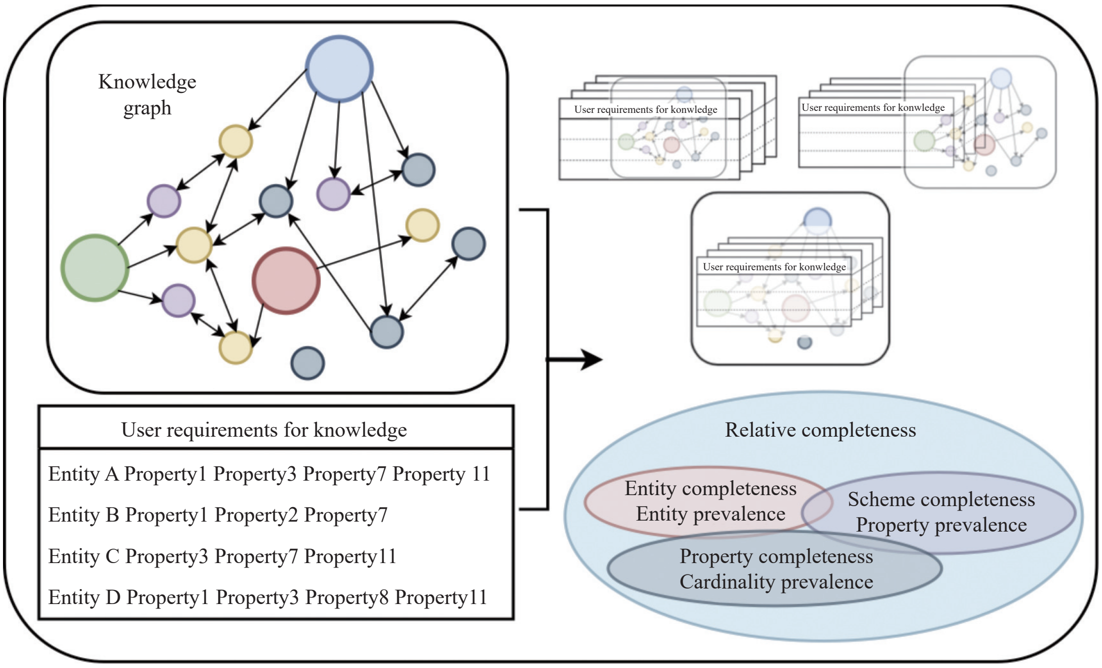


Fig. 3 Knowledge graph completeness measurement

2.3.1 Entity completeness

Entity completeness refers to the ratio of the number of entities in the knowledge graph to the number of entities that meet the user's query requirements. The search volume of users in a certain period is used here as the actual usage demand of users for different entities. In the actual calculation, the click rate or search rate of users is used to approximate the entity prevalence. For example, in a search log dataset, the total search volume is 1 000, and the search volume of a certain entity A is 225, then the search rate of entity A is $\frac{\text{the search volume of entity } A}{\text{total search volume}} = \frac{225}{1\,000}$, and the entity prevalence of entity A is also $\frac{225}{1\,000}$. In the actual calculation process, firstly, the correct calculation formula is selected against the three cases mentioned in Subsection 2.2, and the result obtained is the entity completeness assessment based on user requirements.

2.3.2 Schema completeness

Schema completeness refers to the proportion of classes or properties contained in the knowledge graph that satisfy the user's query requirements. In practice, the property prevalence is approximated by using the click rate, search rate, or preference of users for the properties. For example, if there are five properties, property a, property b, property c, property d, and property e, and the search volume of property b is 650 and the total property search volume is 1 000, then the search rate of property b is $\frac{\text{the search volume of property } b}{\text{total search volume}} = \frac{650}{1\,000}$, and the property prevalence of b is also $\frac{650}{1\,000}$. The schema completeness is finally obtained by extending the numerator part of (1) or (2), depending on the case, and the user's choice of properties. Its numerator part N_{Scheme} is calculated as

$$N_{\text{Scheme}} = \sum_{i=0}^{n} (e_i \cdot p_i) \tag{3}$$

where e_i is the entity prevalence of i, $e_i = \frac{s_i}{S}$, s_i is the search volume of entity i, and S is the total search volume of the entity by the user and p_i is the sum of the property prevalence corresponding to the existence of entity e_i. $p_i = \sum_{j=0}^{m} pp_j$, where $m = |P_{\text{user}} \cap P_{\text{KG}}|$, P_{user} is the property to be queried by the user, and P_{KG} is the property existing in the knowledge graph. pp_j is the property prevalence of j, $\text{pp}_j = \begin{cases} \frac{\text{sp}_j}{\text{SP}}, & \text{property } j \text{ exists} \\ 0, & \text{else} \end{cases}$, where sp_j is the search volume of property j and SP is the total search volume of properties by users.

2.3.3 Property completeness

Property completeness refers to the ratio of the number of property values of a particular property under a certain entity in the knowledge graph to meet the number of property values demanded by users' queries. In the calculation of property completeness, the cardinality assumption is usually used to set a lower bound K for the number of property values, and the corresponding K value is set by counting the user demand for the number of properties with more than one property value when considering the user's query requirements. For example, there exist five properties, property a, property b, property c, property d, and property e. Among them, property a, property b, and property c are properties with unique values such as gender, age, and ID number, so they can be expressed as Card (property a) = Card (property b) = Card (property c) = Card1, respectively. However, property d and property e are properties containing multiple property values, such as company director and movie star, respectively, which require the user to constrain the number of property values according to the actual situation and can be expressed as Card (property d) = Card3 and Card (property e) = Card5. Same with schema completeness, the numerator part of the property completeness, N_{property}, is calculated as

$$N_{\text{property}} = \sum_{i=0}^{n}\left(e_i \cdot \left(\sum_{j=0}^{m} \text{pp}_j \cdot z_{ij}\right)\right) \tag{4}$$

where z_{ij} is used to indicate whether the property value of property j of entity i satisfies the user-defined value of K,

$$z_{ij} = \begin{cases} 1, & \text{the number of property values of property } j \\ & \text{of entity } i \geqslant K_j \\ 0, & \text{else} \end{cases}$$

and K_j is the K value corresponding to property j.

3. Completeness assessment program

In summary, the following will provide a systematic overview of the work of assessing the completeness of the knowledge graph centred around the needs of the users.

3.1 Entity completeness

In the completeness assessment program, entity completeness needs to be calculated first. The relevant entities are searched in the knowledge graph according to the user's usage requirements, and the schema completeness and property completeness are specifically calculated based on the searched entities. For entity completeness, it is first necessary to determine the entity prevalence based on the query frequency of the user for the entity, which can be obtained from the publicly released user browsing dataset. For example, Wikipedia's page view data can be obtained from https://dumps.wikimedia.org/other/pagecounts-raw/website. The specific calculation method needs to refer to the three different cases mentioned in Subsection 2.2 for subsequent completeness assessment.

3.2 Schema completeness

With the entities determined, the schema completeness is calculated for the entities that the user wants to search. First, it is also required to obtain the property prevalence of different properties. The popularity of properties, like the popularity of entities, can be specifically determined from the user's search history. Meanwhile, since the number of property types is much smaller than the number of entities, the popularity of different properties can also be obtained through the user's preference, and the subsequent specific calculations refer to (3).

3.3 Property completeness

In the case of determining properties, it is necessary to set corresponding constraints on the number of property values for these properties, which means using the cardinality assumption to set K-values for different properties that meet user requirements. Since some properties only have unique property values, the K-values of such properties are not discussed. We mainly focus on the properties with multiple property values and determine the appropriate K-values for them through the user's independent choice. After the K-values are determined, the specific calculation can be performed by referring to (4).

Finally, following the mentioned assessment steps, the completeness assessment report can be obtained as shown in Fig. 4, the first three columns are sorted and displayed by the coarseness of the assessment granularity, and the first column represents a list of entities that can be searched in the knowledge graph according to the user's needs, the second column represents a list of relations or properties that can be searched for the corresponding entities, the third column represents the number of tail entities or the number of property values that can be searched, and the fourth column represents the results of the assessment of the various completeness metrics through the computation of the current knowledge graph. More specifically, for example, a user may search for entity e_3 in the knowledge graph KG_{Movie}, and the corresponding relations or properties that may be searched for entity e_3 according to the user's needs are P_1, P_2, P_4, $\cdots$, P_m, $\cdots$, P_q, and the number of tail entities or the number of property values that exist under these relations or properties are 1, 1, 1, $\cdots$, 3, $\cdots$, 1. Eventually, entity completeness is 0.781, schema completeness is 0.756 and property completeness is 0.756. Since the calculation methods of our completeness metrics are nested sequentially in the order of entities, properties, and the number of property values, the assessment results of property completeness can be regarded as the most comprehensive fine-grained completeness assessment results for the knowledge graph to be tested based on the user's needs. The calculation results of entity completeness and schema completeness can be regarded as relatively coarse-grained completeness assessment results.

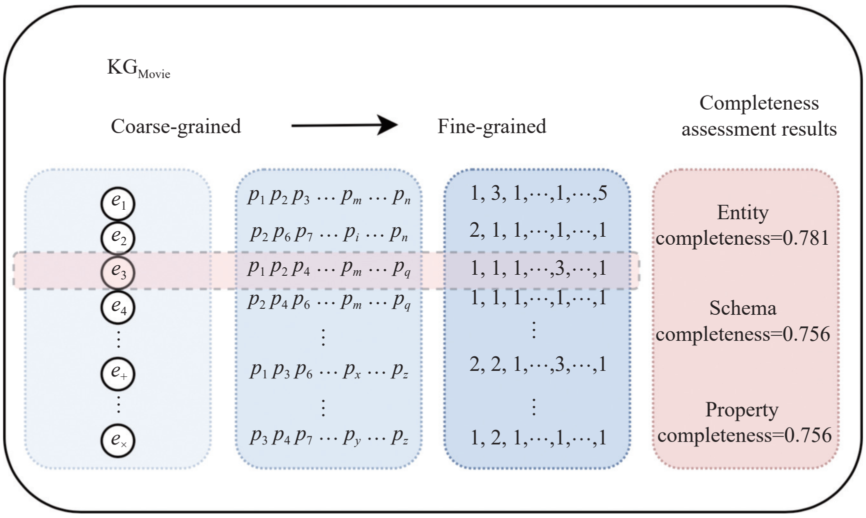


Fig. 4 Example of a movie knowledge graph completeness report

4. Completeness assessment case

The Douban movie knowledge graph and the 1 000 most popular movies knowledge graph will then be specifically assessed for the three integrity indicators in accordance with the assessment scheme.

4.1 Dataset

Users' search records will directly use the readily available dataset of user searches, published in 2015 by Yoshida et al. [33], which counts the number of user searches for movies on the Japanese Wikipedia between 2008 and 2014. Therefore, the relative completeness based on user needs to be assessed refers to the proportion of movies in the knowledge graph that can be searched for by users in terms of the number of movies that users have searched for between 2008 and 2014. In addition, the calculated entity popularity only represents the entity popularity in this timeframe, and the final completeness assessed on the knowledge graph to be tested will also serve as a reference mainly for users in this period. In addition, if we want to obtain more detailed and accurate completeness results for a certain period, we can count the search data of users with smaller time intervals. The search dataset counted by Yoshida et al. is shown in Fig. 5, where the first column is the movie's ID in Wikipedia, the second column is the keyword that the user used to search for the movie, the third column is the movie's complete name in Wikipedia, and the fourth column is the total number of browsers of the corresponding movie for the period. Fig. 6 shows a graph of movie search trends. The horizontal axis represents the entity IDs arranged in descending order, based on search volume, while the vertical axis represents the cumulative search volume of the corresponding entity during the period. The highest search volume is 14 340 020 times, while the lowest is 11 997 times. The difference between the highest and the lowest search volumes is 1 195 times.

2047223	Attack on Titan	Attack on Titan	14 340 020
1426903	Fate/stay night	Fate/stay night	10 576 911
918575	The prince of tennis	The prince of tennis	7 793 509
115545	Boys over flowers	Boys over flowers	6 603746
813729	Black butler	Black butler	6 444475
18961	Fist of the north star	Fist of the north star	6 176 271
5046	Spirited away	Spirited away	5 930 161

Fig. 5 Part of user query dataset

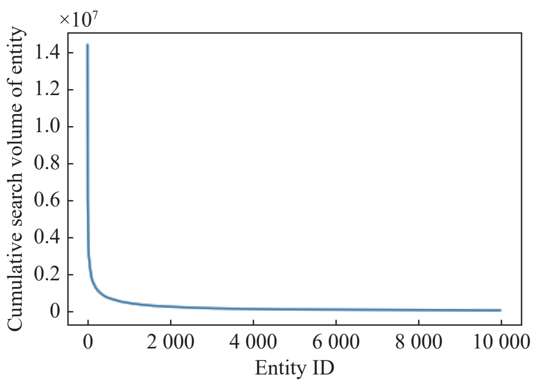


Fig. 6 Trends in movie entity search volume

For other application scenarios, it is also possible to obtain relevant user search data, firstly by following the query methodology shown in https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service/queries/examples/en to obtain all the entities recorded so far under a certain category/domain. For example, the "?item wdt:P31 wd:Q11424" and "?item wdt:P577 ?pubdate" statements can be used to obtain all entities under the category film(Q11424) within a specified time range. Next, the number of searches is counted for the obtained entities in turn. By modifying the entity_name, start_time, and end_time fields of https://wikimedia.org/api/rest_v1/metrics/pageviews/per-article/de.wikipedia/all-access/all-agents/entity_name/monthly/start_time/end_time, the total number of searches for the corresponding entity in the specified time range can be obtained.

The knowledge graphs to be tested are the Douban movie knowledge graph and the 1 000 most popular movies knowledge graph counted in 2021. The Douban movie knowledge graph contains 14 properties including movie ID, movie name, movie URL, cover image, rating, director, composer, actor, movie genre, release region, language, release time, movie duration, and other names, for a total of 4 587 movies. The 1 000 most popular movies knowledge graph contains five properties of movie language, movie popularity, release date, movie rating, and movie genre.

4.2 Calculation process and results

We evaluate the completeness of the to-be-tested knowledge graph based on the first 1 000 search records (about one-half of the total searches) and all search records, respectively. In the following table, 'Top 1 000' refers to 1 000 most popular movies knowledge graph. The completeness assessment results calculated from the first '1 000 search records are shown in Table 2, and the completeness assessment results calculated from all search records are shown in Table 3. The assessment results all show that Douban has the highest entity completeness and is nearly twice as high as the most popular movie knowledge graph. The reason why the results based on all search records are a bit lower than the completeness results using only the first 1 000 search records is that the number of movies that can be searched for remains basically unchanged when the number of search records increases, which is equivalent to the denominator of (1) and (2) increasing while the numerator remains basically unchanged. Ultimately, search tasks similar to this period can yield better search results with the Douban knowledge graph, based on the search dataset provided by Yoshida et al.

Table 2 Completeness calculation results (top 1 000 search results)

Knowledge graph	Metrics		
	Entity completeness	Schema completeness	Property completeness
Douban	0.160 82	0.159 78	0.159 78
Top1 000	0.085 78	0.029 30	0.029 30

Table 3 Completeness calculation results (all search results)

Knowledge graph	Metrics		
	Entity completeness	Schema completeness	Property completeness
Douban	0.119 02	0.121 99	0.121 99
Top1000	0.079 22	0.027 18	0.027 18

For the movie knowledge graph, we can collect the movie-related properties through the Wikipedia page about movies, and eventually get a total of 28 common properties as below. Among them, the properties that commonly have one property value are director, assistant director, producer, executive producer, assistant producer, distributor, film length, origin, language, release date, box office, shooting time, theme song, movie poster, and rating, properties that commonly have one or more values are producer, singer, composer, movie genre, sequel/prequel, location, and award, and properties that commonly have multiple values are screenwriter, actor, cinematographer, voice actor, translation, and soundtrack. Then, based on the user's preference for each property, the following properties can be obtained in order of their popularity: actor, director, movie genre, rating, release date, length, origin, language, sequel/prequel, box office, awards, singer, theme song, and screenwriter. Eventually, the results of the schema completeness assessment can be obtained by calculation as shown in Table 2 and Table 3. Consistent with the assessment results of entity completeness, the same trend is shown in the two different search situations, while Douban is to a large extent more in line with the user's search needs compared to the most popular movie knowledge graph. The assessment of property completeness requires further determination of the corresponding K-values for the properties required by the users. Ultimately, since both knowledge graphs provide the number of property values that satisfy the user's needs, the obtained property completeness assessment results are consistent with the schema completeness assessment results.

5. Discussion

From the evaluation results, it can be seen that the completeness of the two knowledge graphs is not satisfactory, and the main reason for this is that the search records are based on the Japanese Wikipedia, the target users are Japanese, and the movies they mainly search for are also Japanese local movies, so the users' search needs do not match well with the movies contained in the two to-be-tested knowledge graphs. Second, the temporal match between the search records and the two to-be-tested knowledge graphs is also not high. The search dataset counts the user search records between 2008 and 2014, and the two to-be-tested knowledge graphs are constructed in 2019 and 2021, respectively. Therefore, it is usually difficult to get good completeness results when the user's search data and the to-be-tested datasets are not created at the same time. In addition, since the movie domain is inherently and continuously updated, compared to other domains, knowledge graphs in the movie domain that are not steadily updated will have decreasing completeness assessment results. Ultimately, our experimental results also confirm the previously stated conclusion that completeness is subject to change with the user's needs, as well as with the change of time and the updating of knowledge.

Therefore, when selecting a suitable knowledge graph, to select the one that meets the current task, it is generally necessary to combine the needs of the specific application, the update of the domain knowledge, the degree of time matching, the data coverage, and in more detail, it is necessary to consider the amount of data, the existence of specific properties, the number of property values, and so on. In this way, the knowledge graph that meets the current needs can be selected to better achieve the task objectives.

Furthermore, compared with other completeness assessment processes, relative completeness fundamentally involves the consideration of users' actual requirements and the mining of users' search trends, through which a comprehensive result about the completeness dimensions of the knowledge graph is generated, allowing users to select more appropriate knowledge graphs according to their requirement. Also, regular periodic assessments can help users grasp the completeness and lack of knowledge in the knowledge graph and at the same time help knowledge graph builders optimize and improve the construction and updating process of the knowledge graph promptly. At present, many related studies separate assessment and enhancement work, but in fact, it is two parts of a tandem cycle, quality assessment needs to pave the way for quality enhancement, as accurately as possible to find out the places that can greatly improve user satisfaction, the effect of quality enhancement in turn needs to be identified by a suitable and reliable assessment process. Therefore, only when these two parts are used together, can the knowledge graph be put into use quickly and efficiently.

6. Conclusions

We describe a knowledge graph completeness assessment program that incorporates user requirements. The overall completeness level of the knowledge graph is evaluated by three completeness metrics: entity completeness, schema completeness, and property completeness. Through a movie search use case, we evaluate the completeness of the related movie knowledge graphs. Based on the user requirements, we find that a high completeness knowledge graph not only requires a certain scale but also needs to consider changes in user requirements and perform timely update operations. In addition, our completeness assessment program is tailored to the specific needs of each user, considering entities, property categories, and the number of property values and using formulas to calculate completeness results for each metric. Through this process, we can identify missing knowledge in the current knowledge graph based on the user's requirements, allowing us to fill in the necessary target triples. By completing these target triples, we can improve the completeness of the knowledge graph in a time-saving and efficient way to better meet the users' actual needs. Therefore, our future work will be based on the results of the completeness assessment to monitor the completeness of the knowledge graph promptly, so that it can greatly satisfy the needs of users while improving the completeness of the knowledge graph.

References

[1] SHEN T, ZHANG F, CHENG J W. A comprehensive overview of knowledge graph completion. Knowledge-Based Systems, 2022, 255: 109597.

[2] XUE B C, ZOU L. Knowledge graph quality management: a comprehensive survey. IEEE Trans. on Knowledge and Data Engineering, 2023, 35(5): 4969–4988.

[3] PRASOJO R E, DARARI F, RAZNIEWSKI S, et al. Managing and consuming completeness information for wikidata using COOL-WD. Proc. of the 7th International Workshop on Consuming Linked Data, Co-located with the 15th International Semantic Web Conference, 2016. https://ceur-ws.org/Vol1666/paper-02.pdf.

[4] FARBER M, BARTSCHERER F, MENNE C, et al. Linked data quality of dbpedia, freebase, opencyc, wikidata, and yago. Semantic Web, 2018, 9(1): 77–129.

[5] HOGAN A, BLOMQVIST E, COCHEA M, et al. Knowledge graphs. ACM Computing Surveys, 2021, 54(4): 1–37.

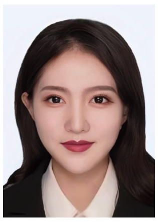
[6] BRACK A, HOPPE A, STOCKER M, et al. Analysing the requirements for an open research knowledge graph: use cases, quality requirements, and construction strategies. International Journal on Digital Libraries, 2022, 23(1): 33–55.

[7] WANG X Y, CHEN L Z, BAN T Y, et al. Knowledge graph quality control: a survey. Fundamental Research, 2021, 1(5): 607–626.

[8] ISSA S, ADEKUNLE O, HAMDI F, et al. Knowledge graph

completeness: a systematic literature review. IEEE Access, 2021, 9: 31322–31339.
[9] LAJUS J, SUCHANEK F M. Are all people married? Determining obligatory attributes in knowledge bases. Proc. of the World Wide Web Conference, 2018: 1115–1124.
[10] BALARAMAN V, RAZNIEWSKI S, NUTT W. Recoin: relative completeness in wikidata. Proc. of the Web Conference, 2018: 1787–1792.
[11] CAPPIELLO C, NOIA T D, MARCU B A, et al. A quality model for linked data exploration. Proc. of the International Conference on Web Engineering, 2016: 397–404.
[12] WISESA A, DARARI F, KRISNADHI A, et al. Wikidata completeness profiling using proWD. Proc. of the 10th International Conference on Knowledge Capture, 2019: 123–130.
[13] BARONCINI S, SARTINI B, VAN ERP M, et al. Is dc: subject enough? A landscape on iconography and iconology statements of knowledge graphs in the semantic web. Journal of Documentation, 2023, 79(7): 115–136.
[14] RAZNIEWSKI S, ARNAOUT H, GHOSH S, et al. Completeness, recall, and negation in open-world knowledge bases: a survey. ACM Computing Surveys, 2023, 56(6): 150.
[15] PELLISSIER T T, STEPANOVA D, RAZNIEWSKI S, et al. Completeness-aware rule learning from knowledge graphs. Proc. of the 16th International Semantic Web Conference, 2017: 507–525.
[16] GALARRAGA L, RAZNIEWSKI S, AMARILLI A, et al. Predicting completeness in knowledge bases. Proc. of the 10th ACM International Conference on Web Search and Data Mining, 2017: 375–383.
[17] DARARI F, RAZNIEWSKI S, PRASOJO R, et al. Enabling fine-grained RDF data completeness assessment. Proc. of the International Conference on Web Engineering, 2016: 170–187.
[18] LUGGEN M, DIFALLAH D, SARASUA C, et al. Non-parametric class completeness estimators for collaborative knowledge graphs — the case of wikidata. Proc. of the International Semantic Web Conference, 2019: 453–469.
[19] CHERIX D, USBECK R, BOTH A, et al. CROCUS: cluster-based ontology data cleansing. Proc. of the CEUR Workshop, 2014: 7−14.
[20] SOULET A, GIACOMETTI A, MARKHOFF B, et al. Representativeness of knowledge bases with the generalized Benford’s law. Proc. of the International Semantic Web Conference, 2018: 374–390.
[21] SIMSEK U, KARLE E, ANGELE K, et al. A knowledge graph perspective on knowledge engineering. SN Computer Science, 2022, 4(1): 16.
[22] ABIAN D, MERONO-PENUELA A, SIMPERL E. An analysis of content gaps versus user needs in the wikidata knowledge graph. Proc. of the International Semantic Web Conference, 2022: 354–374.
[23] SYCHEV O. Combining neural networks and symbolic inference in a hybrid cognitive architecture. Procedia Computer Science, 2021, 190: 728–734.
[24] JI G L, LIU K, HE S Z, et al. Knowledge graph completion with adaptive sparse transfer matrix. Proc. of the AAAI Conference on Artificial Intelligence, 2016. DOI: https://doi.org/10.1609/aaai.030i1.10089.
[25] AKINNUBI A, AJIBOYE J. Knowledge graph: a survey. Journal of Robotics and Automation Research, 2023, 4(2): 366–377.
[26] CHEN X J, JIA S B, XIANG Y. A review: knowledge reasoning over knowledge graph. Expert Systems with Applications, 2020, 141(112): 948.
[27] PAN S R, LUO L H, WANG Y F, et al. Unifying large language models and knowledge graphs: a roadmap. IEEE Trans. on Knowledge and Data Engineering, 2024. DOI: 10.1109/TKDE.2024.3352100.
[28] TIWARI S, Al-ASWADI F N, GAURAV D. Recent trends in knowledge graphs: theory and practice. Soft Computing, 2021, 25(31): 8337–8355.
[29] CUI Y N, WANG Y X, SUN Z Q, et al. Lifelong embedding learning and transfer for growing knowledge graphs. Proc. of the AAAI Conference on Artificial Intelligence, 2023, 37(4): 4217–4224.
[30] TAMASAUSKAITE G, GROTH P. Defining a knowledge graph development process through a systematic review. ACM Transactions on Software Engineering and Methodology, 2023, 32(1): 1–40.
[31] CHEN Z, WANG Y H, ZHAO B, et al. Knowledge graph completion: a review. IEEE Access, 2020, 8: 192435–192456.
[32] WANG S, HUANG X, CHEN C, et al. Reform: error-aware few-shot knowledge graph completion. Proc. of the 30th ACM International Conference on Information & Knowledge Management, 2021: 1979−1988.
[33] YOSHIDA M, ARASE Y, TSUNODA T, et al. Wikipedia page view reflects web search trend. Proc. of the ACM Web Science Conference, 2015: 65.

Biographies



ZHANG Ying was born in 1996. She received her M.S. degree in data science from King’s College of London, British, in 2020. She is working toward her PH.D. degree in the Institute of Systems Engineering, Academy of Military Sciences. Her research interests include knowledge graph refinement and knowledge graph evaluation.
E-mail: yingzhang199608@foxmail.com

XIAO Gang was born in 1973. He received his B.S. and Ph.D. degrees in computer science and technology from the National University of Defense Technology. He is currently a professor at the National Key Laboratory for Complex Systems Simulation. His research interests include complex system theory, system design, knowledge graphs, and deep learning.
E-mail: searchware@qq.com