

# KBMetrics – A Multi-purpose Tool for Measuring the Quality of Linked Open Data Sets

Tong Ruan, Xu Dong, Yang Li, and Haofen Wang

East China University of Science & Technology, Shanghai, 200237, China  
ruantong@ecust.edu.cn, dongxu0220@qq.com, marine1ly@163.com, whfcarter@ecust.edu.cn

**Abstract.** While several quality assessment tools focus on evaluating the quality of Linking Open Data (LOD), most tools fail to meet diverse quality assessment requirements from the users' perspective. In this demo, we categorized quality assessments requirements into three layers: *understanding* the characteristics of data sets, *comparing* groups of data sets, and *selecting* data sets according to user-defined usage scenarios. We have designed KBMetrics to incorporate the above quality assessment purposes. Not only does the tool incorporate different kinds of metrics to characterize a data set, but it has also adopted ontology alignment mechanisms for comparison purposes. Most importantly, end users can define usage contexts to adapt to different usage scenarios. Both the quality assessment processes and findings in these data sets show the effectiveness of our tool.

## 1 Introduction

In recent years, an increasing number of semantic data sources have been published on the Web. There is great demand for knowledge about the qualities of these data sets. Several tools target at quality assessment tasks. For example, with Flemming<sup>1</sup> in German, users could get the ultimate quality value of a data set after interactively inputting parameters on lists of metrics. TripleCheckMate<sup>2</sup> is a crowdsourcing quality assessment tool focusing on correctness evaluation of DBpedia. However, these currently available tools fail to meet the diversity requirements of quality assessment. In this paper, we classify the goal of quality assessment into three layers, as shown in the pyramid on the left of Figure 1.

- *Understand* the characteristics of data sets. There exist lots of metrics on evaluating special aspects of LOD qualities, including data size, data complexity, and data consistency. The required metrics vary according to the data sets. For example, web-scale extracted data sets such as DBpedia and YAGO are prone to errors, so that the Correctness Ratio metric is under investigation. On the other hand, for domain-oriented human constructed data sets published in LOD, for example, Drugbank, the number of instances in the domain may be of great importance.
- *Compare* different data sets. The quality of a data set could be better understood by comparing its evaluation metrics values with those of other data sets. For example, users may have no idea about the meaning of instance size 100,000, while they could easily understand that one data set is larger than

<sup>1</sup> <http://linkeddata.informatik.hu-berlin.de/LDSrcAss/datenquelle.php>

<sup>2</sup> <http://aksw.org/Projects/TripleCheckMate.html>

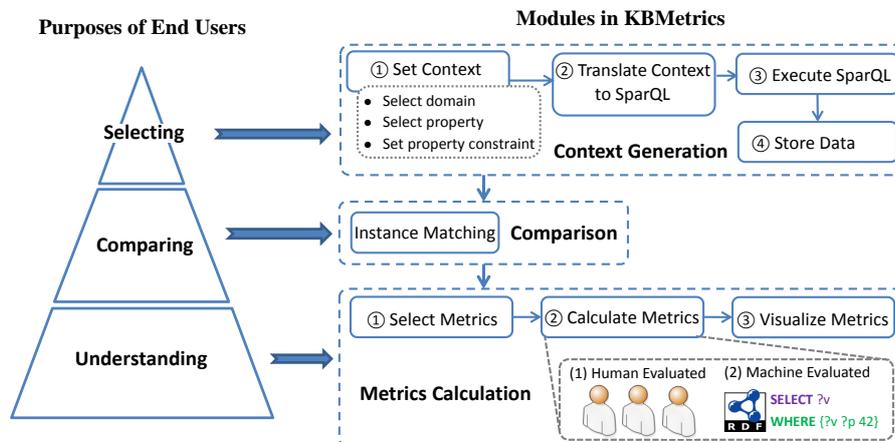


Fig. 1. Functions in KBMetrics and its Relation with Purposes of Quality Assessment

another. Furthermore, the comparisons become more meaningful if they are carried out under the same or similar conditions. For instance, it is better to compare Drugbank and DBpedia on the drug-related domain instead of all other domains defined in DBpedia. In that case, calculating the metrics on the overlapped instances or the overlapped domains is fairer and more reasonable.

- *Select* suitable data sets. The ultimate goal of quality assessment is to help end users determine which data sets are “fit for use” for their data usage requirements. Traditionally, data quality is commonly conceived of as “fitness of use for a certain application or user case”. For example, as mentioned in <sup>3</sup>, “DBpedia currently can be appropriate for a simple end-user application but could never be used in the medical domain for treatment decisions”. However, the questions of how to define the “Usage Contexts” and how to link these contexts to the quality assessment process have not been well investigated in the literatures. To the best of our knowledge, there exists no tool that can let users adapt the quality assessment process to their usage scenarios.

In this demonstration, we present a multi-purpose tool, KBMetrics, for the quality evaluation of Linked Open Data Sets. The tool can support the three purposes mentioned above. We also apply corresponding evaluation processes to DBpedia and YAGO.

## 2 Modules in KBMetrics

The relationship between functions in KBMetrics and the three purposes mentioned above is shown in Figure 1.

- *Understanding*: The understanding purpose is transformed into the *Metrics Calculation* module. Users can *Select Metrics*, *Calculate Metrics*, *Visualize Metrics Results*, and *Compare/Analyze results*, as shown in Figure 1. The tool has 12 built-in metrics in five categories. The details of the metrics and the methods of calculation can be found via<sup>4</sup>. The tool can not only sup-

<sup>3</sup> <http://ldq.semanticmultimedia.org/cfp>

<sup>4</sup> <http://kbeval.nlp-bigdatalab.com/docs/doc.pdf>

port machine-computable metrics such as data size, but it can also support human-evaluated metrics such as correctness. We store data in Jena and pose SparQL queries to get the answers of *Machine-Computable* metrics, and also design a process for human-evaluated metrics. The process includes sampling a sub-data set to lessen human efforts, assigning tasks to more than three evaluators to reduce individual subjective impacts, resolving inconsistencies between different evaluators, and calculating the result. Currently, the tool supports two sampling methods, random sampling and the Wilson Interval Score<sup>5</sup>.

- *Comparing*: If end users want to calculate metrics on overlapped instances or overlapped domains, the additional schema alignment and instance matching module is provided. Both schema alignment and instance matching belong to the scope of ontology alignment, which has been studied for years. Moreover, the community of ontology alignment provides sufficient tools so that our module mainly provides interfaces to integrate the results of a third-party ontology alignment tool (i.e., PARIS). The results are represented as triples with the predicates “owl:sameAs”, “owl:EquivalentClass”, or “owl:EquivalentProperty”.
- *Selecting*: A pre-processed step to filter users’ requirements is supported by the *Context Calculation* module. Four steps are required to fulfill context calculation: a) *Define the Context*: Users can input their data requirements based on their usage scenarios with GUI interfaces. We support various types of contexts, e.g. the *Domains Context* (such as *cities*, and *organizations*), the *Property Context* (such as *populations of cities*), or the *Property Constraint Context* (such as *the presidents of the USA* or *the 100 biggest cities in the world*). b) *Context Transformation*: The context definitions on the UI are translated into executable SparQL queries. The queries may be different in different data sets due to the vocabulary differences. c) *Context Execution*: The queries are executed on target data sets. d) *Store Data*: The results under contexts, namely the sub-data sets, are stored in Jena too. Users may perform metrics evaluation on the sub-data sets.

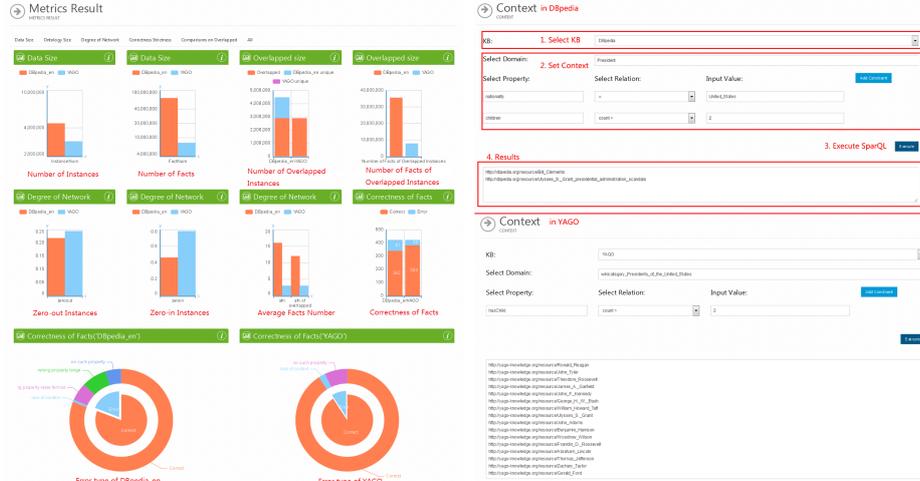
### 3 Demonstration

Our demonstration contains three typical scenarios. A recorded video of KBMetrics can be downloaded at <http://kbeval.nlp-bigdatalab.com/iswc2015.wmv>, and the system can be accessed online as well<sup>6</sup>.

**Evaluate A Single Data Set** In this demonstration, we firstly select DBpedia as a target KB and metrics such as Data Size and Degree of Network. We find that the 2014 version of DBpedia has 4,465,631 instances and 68,112,887 facts. We further select the Correctness metric and a GUI interface to let users select sampling methods, and related parameters appear. After we choose the default parameters, the system randomly selects 423 samples from DBpedia according to sampling theory. Then we assign tasks to different evaluators. After

<sup>5</sup> [http://en.wikipedia.org/wiki/Binomial\\_proportion\\_confidence\\_interval](http://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval)

<sup>6</sup> <http://kbeval.nlp-bigdatalab.com/v12/>



**Fig. 2.** Compare DBpedia with YAGO in **Fig. 3.** Context Definition and Execution in KBMetrics

each evaluator successfully evaluates each data item, the system gives the ultimate correctness ratio result, 0.81. However, we have no idea whether 0.81 is good or bad, or 4,465,631 instances is large or small. Therefore we add YAGO for comparison, and the version of YAGO is YAGO2s.

**Compare Multiple Data Sets** Figure 2 shows that DBpedia is richer than YAGO in data size. The number of instances in YAGO is half that in DBpedia, and the number of facts in YAGO is about a tenth of that in DBpedia. Through the results of the overlapped metrics we find that DBpedia almost contains all of the instances in YAGO. The average number of facts each overlapped instance has in YAGO is 3, which is about the same as that of the whole instances in YAGO. However in DBpedia, the average number of overlapped instances is slightly smaller than that of the whole instances. So the distribution of facts on overlapped instances in DBpedia is different from that on whole instances. The Degree of Network shows that the connections between DBpedia instances are richer than those between YAGO instances. But the correctness of YAGO is 0.91 from our evaluation, and it is greater than that of DBpedia.

**Select Data Sets on User Context** From the above, we may find DBpedia to be richer than YAGO. However, it is not the case in special user contexts. For example, we want to conduct a survey on *Presidents of the United States having more than two children*. In DBpedia, we set the domain as “President”, and add two constraints, including the “nationality” and the number of “children”. We directly set the domain as “Presidents of the United States”, since YAGO has a richer taxonomy system. After adding the number of “hasChild” constraints in YAGO, 16 presidents returned as shown in Figure 3. By contrast, DBpedia has 2 presidents returned. The reason is that, although DBpedia contains all those instances in YAGO, many of them do not belong to “President” type. Furthermore, DBpedia has many properties denoting the same relationship, for instance, “country” and “nationality”, and it does not consolidate these relationships.