WC3: Wikipedia Category Comprehensiveness Checker based on the DBpedia metadata database

Masaharu Yoshioka[1]

Graduate School of Information Science and Technology, Hokkaido University
N14 W9, Kita-ku, Sapporo 060-0814, Japan

Abstract. We demonstrate Wikipedia Category Comprehensiveness Checker (WC3) based on the DBpedia metadata database. This system supports to check comprehensiveness and consistency of Wikipedia category annotation using DBpedia database. This system is available online.

1 Introduction

Wikipedia (`http://www.wikipedia.org/`) is a free, Wiki-based encyclopedia maintained by many volunteer editors. Because it includes many articles, Wikipedia categories are used to find appropriate articles for particular interests. Methods for constructing Wikipedia ontologies, such as YAGO2 [1] and a Japanese Wikipedia ontology [2], use Wikipedia categories to construct semantic hierarchies and class–instance relationships. Despite the importance of the Wikipedia categories, there are no good systematic methods or tools to support or maintain the comprehensiveness of the category annotation.

In this paper, we propose a Wikipedia Category Comprehensiveness Checker (WC-triple or WC3: formerly named as Wikipedia Category Consistency Checker[3]) that has a SPARQL query database for representing Wikipedia categories using DBpedia [4] metadata and their analysis results. By using this system, the user can check the comprehensiveness of the category (e.g., percentage of candidate Wikipedia pages for the category that are annotated; percentage of Wikipedia pages for the category that have appropriate infobox information). WC3 also uses DBpedia Live to check the effectiveness of recent edits to the pages in the category.

2 Automatic SPARQL Construction by WC3

WC3 [3] aims to analyze set-and-topic-style Wikipedia categories (e.g., "Cities_in_France": "Cities" is a set and "France" is the topic) by constructing appropriate SPARQL queries that combine two restrictions. The first applies to the set and is represented by using a type predicate (rdf:type). The second restriction applies to the topic. WC3 generates restriction candidates by combining those candidates as candidate queries.

All candidates are evaluated based on comparisons between the retrieved results of the query and articles that belong to the category. There are four types of articles for the query. Articles that belongs to a target category are classified into two types; Found (retrieved by the query) and NotFound (not retrieved). Retrieved results that are not belong to the target category are classified into two types; ChildrenError (retrieved articles that belong to children categories and

Error (other retrieved articles). Precision, recall, and f-measures are calculated by Found/(Found+Error), Found/(Found+NotFound), and the harmonic mean of precision and recall, respectively.

The system selects the SPARQL query that has the highest f-measure among all candidates. The following is a candidate SPARQL query for "People_from_Tokyo"[1].

```
SELECT DISTINCT ?s
WHERE {?s rdf:type dbo:Person .
?s dbp:birthPlace dbr:Tokyo
MINUS { ?s dbo:wikiPageRedirects ?o . }}
```

There are two problems with the previous WC3. The first is computation time. Because SPARQL query generation requires generating many SPARQL candidates and evaluating their quality, it takes a long time (around 1 min) to obtain the final result. The other is related to freshness. WC3 uses local DBpedia archives, so when the editors modify Wikipedia articles based on WC3's suggestions, WC3 cannot confirm the appropriateness of the editing results.

3 New WC3 System

To solve these problems with the previous WC3, we implemented a newer version by adding the following two modules. This system is available online and the URL of the system is `https://wnews.ist.hokudai.ac.jp/wc3/`. It also has links to a demonstration movie and a detailed help page.

SPARQL query database SPARQL queries generated for Wikipedia categories are cached in the database, and the user can modify a query and store the modified query when it is better than an existing one. Retrieved results are also cached in the database to allow checking the comprehensiveness of the Wikipedia category annotation.

Access to DBpedia Live The system can send the same SPARQL query to DBpedia Live (`http://live.dbpedia.org/`) instead of the local DBpedia server to evaluate the appropriateness of the editing results.

The system has the following functions.

1. Retrieve SPARQL query database
   When the user inputs the name of the Wikipedia category, the system returns SPARQL queries and pre-computed retrieved result are shown as a result. The user can modify the SPARQL query and compare the retrieved results of the original query and the new query. The user can also send the same SPARQL query to DBpedia Live to check the appropriateness of any edits conducted after the preparation of the current DBpedia archive.
2. Check the appropriateness of the Wikipedia category annotation for the page
   The system summarizes the results of the Wikipedia category analysis for the page.
3. SPARQL query construction
   There are two SPARQL query construction methods in the system. One is the automatic SPARQL query construction method used by the previous WC3. The other is the construction of new SPARQL queries based

[1] "dbo," and "dbp," are abbreviations for "http://dbpedia.org/ontology," "http://dbpedia.org/property," respectively

on SPARQL queries for sibling categories. For example, SPARQL queries for "People_from_Nagoya" are constructed by using sibling categories "People_from_Tokyo.". In this case string "Tokyo" in the query are replaced by "Nagoya."

This system uses latest DBpedia database (2016-10 version). Target categories for WC3 are all Wikipedia categories with the following conditions: 1) subcategories of "Categories by parameter" with set-and-topic-style excluding stubs; 2) at least one article directly belongs to the category. There are 655,937 target categories in this version of DBpedia. In this experiment, we selected target categories that have at least 10 articles that directly belong to the category for initial database construction (231,148 categories).

Figure 1 shows a screen-shot of the system. First, the user enters a category name using the "load" button. Then, information about a stored SPARQL query is loaded if one exists. If no stored information is available, the user can generate a query using the automatic SPARQL query construction method of WC3 or by using information from sibling categories. To highlight the quality of the SPARQL query, when the recall and precision are larger than 0.7 or lower than 0.3, the corresponding elements are highlighted in green or red, respectively.

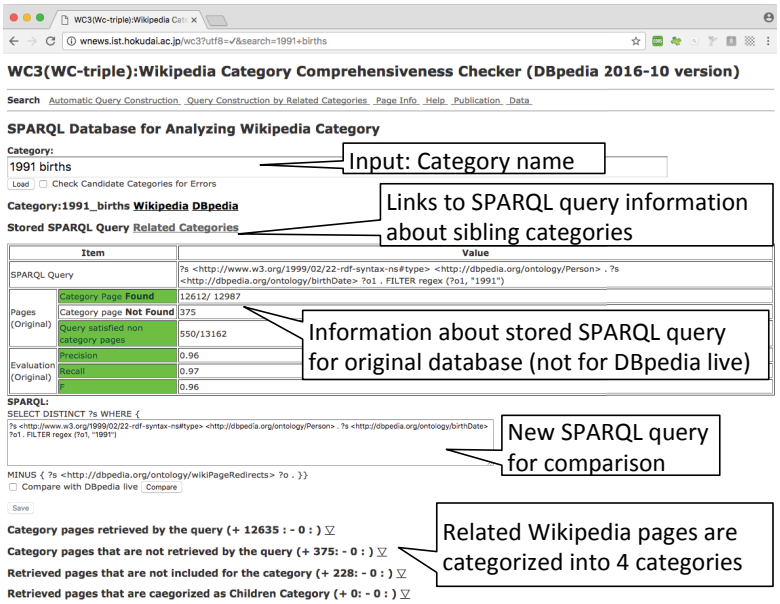

Fig. 1. Screen-shot of WC3 interface

The user can check information about a page in Wikipedia or DBpedia page information stored in the database by clicking the corresponding links. When the user adds a check-mark to the "Check Candidate Categories for Errors" box, a list of candidate categories is shown as additional columns (Figure 2). For example, candidate categories for "Adam_Jezierski" and "Adil_Ibragimov" are "1990_births" and "1989_births"[2]. In both cases, the suggested candidate

[2] All pages were accessed on May. 18, 2018

categories seem to be appropriate for those pages according to the information in the text and the infobox information for the page.

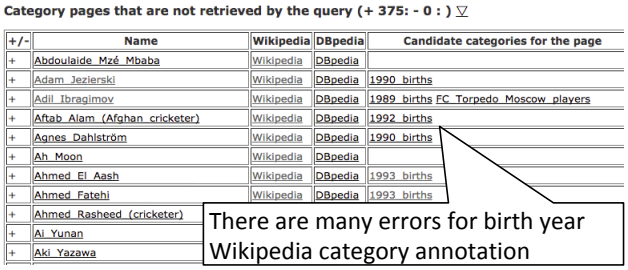

Fig. 2. List of NotFound pages for "1991_births" with candidate categories

The user can also check the appropriateness of the edits conducted after the most recent DBpedia database construction by checking the "Compare with DBpedia Live" box and clicking on the "Compare" button. Comparison results are shown with the same categories for the "load" case. "+" and "–" show pages that can be categorized by using DBpedia Live only or by the original database, respectively. Results from DBpedia Live are also stored for checking the comprehensiveness of the Wikipedia page edits.

4 Conclusion

In this paper, we have introduced a new WC3 system that uses the SPARQL query database and DBpedia Live. This system solves the problems of the previous WC3 (computation time and freshness of the database). This system is available online and supports volunteer editors in maintaining Wikipedia categories. Updating Wikipedia categories based on this framework is also beneficial for knowledge engineers who would like to utilize Wikipedia as a knowledge resource.

5 Acknowledgement

This work was partially supported by JSPS KAKENHI Grant Number 18H03338.

References

1. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. Artificial Intelligence **194**(0) (2013) 28 – 61
2. Tamagawa, S., Sakurai, S., Tejima, T., Morita, T., Izumi, N., Yamaguchi, T.: Learning a large scale of ontology from japanese wikipedia. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. Volume 1. (Aug 2010) 279–286
3. Yoshioka, M. In: WC3: Analyzing the Style of Metadata Annotation Among Wikipedia Articles by Using Wikipedia Category and the DBpedia Metadata Database. Springer International Publishing, Cham (2017) 119–136
4. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - a crystallization point for the web of data. Web Semantics: Science, Services and Agents on the World Wide Web **7**(3) (2009) 154 – 165