# Infoboxer: Using Statistical and Semantic Knowledge to Help Create Wikipedia Infoboxes

Roberto Yus[1], Varish Mulwad[2], Tim Finin[2], and Eduardo Mena[1]

[1] University of Zaragoza, Zaragoza, Spain
{ryus,emena}@unizar.es,
[2] University of Maryland, Baltimore County, Baltimore, USA
{varish1,finin}@cs.umbc.edu

**Abstract.** Infoboxer uses statistical and semantic knowledge from linked data sources to ease the process of creating Wikipedia infoboxes. It creates dynamic and semantic templates by suggesting attributes common for similar articles and controlling the expected values semantically.

**Keywords:** Infoboxes, Wikipedia, DBpedia, Semantic Web

## 1 Introduction

Wikipedia is a free and collaborative encyclopedia launched in 2001 which, as of June 2014, has more than four million English articles. Wikipedia is centered around collaboratively creating and editing articles for a variety of topics and subjects. The information in these articles is often split into two parts: 1) unstructured text with details on the article's subject and 2) a semi–structured *infobox* that summarizes the most important facts about the article's subject. Thus, infoboxes are usually preferred by systems using Wikipedia content (such as Google's Knowledge Graph or Microsoft Bing's Satori) as they are easier to process by machines.

Current creation of Wikipedia infoboxes is based on templates that are created and maintained collaboratively. While templates provide a standardized way of representing infobox information across Wikipedia articles, they pose several challenges. Different communities use different infobox templates for the same category articles; attribute names differ (e.g., date of birth vs. birthdate), and attribute values are expressed using a wide variety of measurements and units [2]. Infobox templates are grouped by article categories with typically one template associated with one category (e.g., it is hard to find an infobox template for article whose categories are both Artist and Politician). Given the large number of Wikipedia categories, it is difficult to create templates for every possible category and combination. Finally, templates are free form in nature; when users fill attribute values no integrity check is performed on whether value is of appropriate type for the given attribute, often leading to erroneous infoboxes.

*Infoboxer*[3] is a tool grounded in Semantic Web technologies that overcomes challenges in creating and updating infoboxes, along the way making the process easier for users. Using statistical information from Linked Open Data (LOD) datasets, Infoboxer helps people populate infoboxes using the most popular attributes used to describe instances for a given category or any combination of categories, thus generating an infobox "template" automatically. For each attribute or property Infoboxer also identifies the most popular types and provides them as suggestions to be used to represent attribute values. The attribute value types allows Infoboxer to enforce semantic constraints on the values entered by the user. It also provides suggestions for attribute values whenever possible and links them to existing entities in Wikipedia.

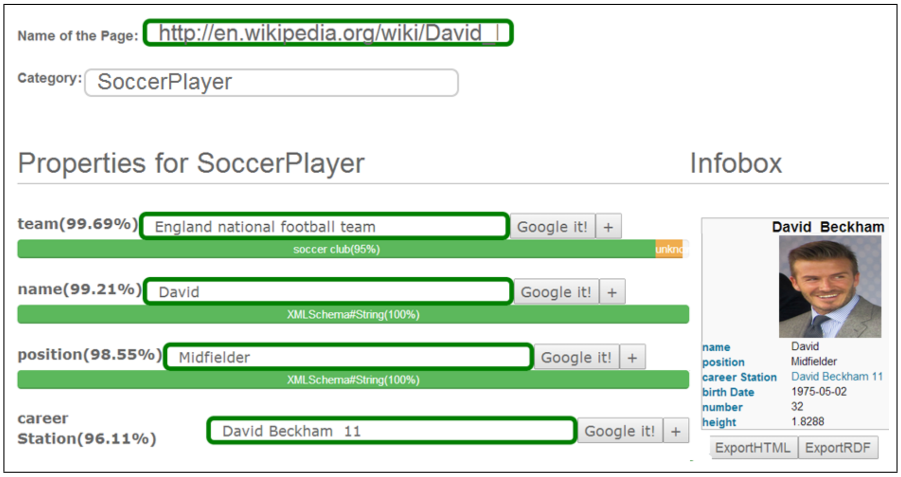
## 2 Using DBpedia to Help Creating Wikipedia Infoboxes

The Infoboxer demonstration presented in this paper, uses DBpedia [1], a semi-structured representation of Wikipedia's content, to implement and power all of its features and functionalities. While our demonstration system uses DBpedia, it could be replaced with any other LOD knowledge base, such as Yago or Freebase. In the following sections we explain each functionality in detail.

*Identifying popular attributes.* The most popular attributes for a given category are generated by computing attribute usage statistics based on instance data for the category. Infoboxer first obtains a list of DBpedia instances for the given category. For example, list of instances associated with the category *dbpedia-owl:SoccerPlayer* include *dbpedia:David_Beckham* and *dbpedia:Tim_Howard*. A list of attributes used by these instances is generated and then ordered based on number of instances using each attribute. Duplicate counts are avoided by noting distinct attribute for every instance only once (at this point we want to know how many different instances of the category are using the property to highlight its popularity). For example, the property *dbpedia-owl:team* appears several times with the soccer player *dbpedia:David_Beckham* (as he played for several soccer teams), but it is only counted once.

Sorting the list of attributes based on frequency of usage provides Infoboxer with the most popular attributes for each category. Figure 1 shows the most popular properties for soccer players, e.g., *dbpedia-owl:team*, *foaf:name*, and *dbpedia-owl:position*, along with the percentage of instances using them. This first step could be simplified by only using information about the domains and ranges of each property (e.g., to obtain properties where the domain is a soccer player). However, DBpedia does not impose restrictions over domain and range for most of the properties. In fact, in a previous analysis, we detected that for DBpedia 3.9, 21% of properties have no domain defined, 15% have no range, and 2% have no domain and range. On July 1, Wikidata, a project focused on human

[3] `http://sid.cps.unizar.es/Infoboxer`



Name of the Page: http://en.wikipedia.org/wiki/David_

Category: SoccerPlayer

Properties for SoccerPlayer

team(99.69%) England national football team

soccer club(95%) unkno

name(99.21%) David

XMLSchema#String(100%)

position(98.55%) Midfielder

XMLSchema#String(100%)

career Station(96.11%) David Beckham 11

Infobox

David Beckham

| | |
|---|---|
| name | David |
| position | Midfielder |
| career Station | David Beckham 11 |
| birth Date | 1975-05-02 |
| number | 32 |
| height | 1.8288 |

**Fig. 1.** Screenshot of Infoboxer creating the Wikipedia infobox of a soccer player.

edited structured Wikipedia, rolled out a similar feature which is restricted to suggesting only popular properties[4].

*Identifying popular range types.* Infoboxer finds the most popular types used to represent values for each attribute identified in the previous step. Attribute value types is akin to *rdfs:range* classes associated with an attribute or a property in an ontology. Infoboxer first obtains a list of attribute values for a given category and attribute by identifying list of triples in DBpedia's ABox whose subject are instances of the given category and property, the given attribute. For example, the category *dbpedia-owl:SoccerPlayer* and attribute *dbpedia-owl:team* generates a list of values such as *dbpedia:Arsenal_F.C.* and *dbpedia:Korea_University*. A list of value types is generated from the values and ordered based on number of instances whose attribute values have the type. Based on the attribute, value types are either semantic classes, such as *dbpedia-owl:SoccerClub* and *dbpedia-owl:University*, or xml datatypes such as *xsd:string*, *xsd:integer*, or *xsd:datetime*. Sorting the list of types provides Infoboxer with the most popular attribute value (or range) types.

*Suggesting attribute values and enforcing semantic constraints.* The top three value types for an attribute are provided as suggestions to users as they add values for the most popular attributes in the infobox. Infoboxer also uses these types to enforce semantic constraints on the values entered, thus ensuring infobox correctness. In cases where value type is a semantic class, Infoboxer retrieves instances of that class and populates them for auto-completion as user starts filling up the value. In cases where value type is an XML datatype, Infoboxer

[4] http://lists.wikimedia.org/pipermail/wikidata-l/2014-July/004148.html

shows the most popular values used as examples. Once the user enters a value, Infoboxer checks whether value conforms to the expected type.

*Fixing existing infoboxes.* Infoboxer also uses its functionalities to improve existing Wikipedia infoboxes. Given an article title, Infoboxer fetches its categories and existing attribute values. Then, it highlights popular properties with missing values and also highlights attribute values that have an incorrect semantic type. For example, as of June 2014, *dbpedia:David_Beckham* has the value *dbpedia:England_national_football_team* (whose *rdf:type* is *dbpedia-owl:SoccerClub*) for the attribute *dbpedia-owl:birthPlace* and Infoboxer highlights it as a possible error as only 2% of soccer players have a soccer club as birth place (49% of them have a *dbpedia-owl:Settlement* and 22% a *dbpedia-owl:City*). Also, Infoboxer encourages users to update the attribute value if it is of a less popular type (e.g., suggesting a value of type *dbpedia-owl:SoccerClub* over *dbpedia-owl:Organisation* for the property *dbpedia-owl:team*).

The combination of the four functionalities allows Infoboxer to dynamically generate infobox templates, ensure infobox correctness, and help assist in fixing existing ones. Since Infoboxer relies on KBs such as DBpedia, generated templates will automatically evolve with change in information in KBs over time.

## 3 Demonstration

The demo will allow users to create new infoboxes and edit existing ones. They begin by entering the name of a new or existing Wikipedia article and select appropriate categories for it (e.g., Soccer Player and Scientist). Users will be provided with the most popular attributes to be completed, along with its popularity, based on the selected categories. For each attribute, users will also be provided information about the top three value types; auto-complete will assist users in selecting the appropriate value. A "Google it" button will help user fire Google search queries to discover a possible value. Also, as users start filling values in the forms, current version of the infobox will be displayed on the side. In summary, users will be able to experience how fast and controlled it is to create semantically correct Wikipedia infoboxes with Infoboxer.

***Acknowledgments.*** This research was supported by the CICYT project TIN-2010-21387-C02-02, DGA FSE, NSF awards 1228198, 1250627 and 0910838 and a gift from Microsoft Research.

## References

1. 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), 154–165 (2009)
2. Morsey, M., Lehmann, J., Auer, S., Stadler, C., Hellmann, S.: DBpedia and the Live Extraction of Structured Data from Wikipedia. Program: electronic library and information systems 46, 157–181 (2012)