International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013

Towards Bengali DBpedia

Arup Sarkar\textsuperscript{a,}\textsuperscript{*}, Ujjal Marjit\textsuperscript{b}, Utpal Biswas\textsuperscript{a}

\textsuperscript{a}Dept. of Computer Sc. & Engg, University of Kalyani, Kalyani 741235, W.B. India
\textsuperscript{b}C.I.R.M., University of Kalyani, Kalyani 741235, W.B. India

Abstract

Online encyclopedia brings a whole universe of real information over a mouse click today. In the recent years, online encyclopedia like Wikipedia shows how big and vast it can be. Collecting information from such a big resource and producing as per the user’s requirement always is a challenging task. Linked Data based project called DBpedia resolves this challenge to some extent by publishing data from Wikipedia semantically. Like Wikipedia, DBpedia also gives its datasets in different international languages. Generation of DBpedia content in different languages always requires some extra effort to resolve the language related issues. Some special configuration and settings to be maintain within the DBpedia framework. This paper explains how a Bengali version can be achieved from the original version of the DBpedia framework.

© 2013 The Authors. Published by Elsevier Ltd. Open access under CC BY-NC license.
Selection and peer-review under responsibility of the University of Kalyani, Department of Computer Science & Engineering

Keywords: DBpedia; Bengali DBpedia; DBpedia Information Extraction Framework; Wikipedia

1. Introduction

Proper use of knowledge can change the history of mankind forever. That is why knowledge always remains a matter of praise of the intelligent ones. Now a day’s web becomes a huge hub of information storage. A place where information gets stored, retrieved and shared. Online encyclopedia has become very famous concept today, since it provides different category of information in a single place. Among them, most well known is the Wikipedia project. We can define Wikipedia as a common place to share up to-date information worldwide. Though Wikipedia holds a lot of structured data but still it is not completely usable because it is represented over the traditional non-semantic web just like a collection of textual data, and it is the end user’s sole responsibility to find out their needs of correct information from thousands of Wiki pages from Wikipedia. The solution is to add semantic annotations to the wiki-pages of Wikipedia. With the methodology of Semantic Wiki it was possible to add the semantic annotations to the pages manually but it was not a scalable solution for a huge datasets like Wikipedia where information is updated every seconds making Wikipedia growing all the time. Manual approach seemed unfit at this stage. So an automated technique was required. DBpedia project, initially known as DBpedia, a joint venture by group of

\textsuperscript{*} Corresponding author.
\textit{E-mail address:} arup@klyuniv.ac.in
researchers from Freie Universitat and University of Leipzig and OpenLink Softwares is just that solution world looking for. DBpedia project represents an important issue regarding the Linked Open Data (LOD) project. LOD mainly deals with freely accessible Semantic Data over the Semantic Web and publishes them as Linked Data[1] over the LOD cloud. It also maintains the links among the different data graphs. Maintaining links between different RDF (Resource Description Framework) [2] statements across different data sets is one of the important aspects of LOD project, since it is dedicated to the Linked Data. To make the LOD project a story of success, DBpedia plays an important role. Till date, a huge number of datasets over the LOD is connected with the DBpedia datasets to become a genuine part of the LOD project. One main reason behind this nature of the activity of the participants of LOD is that they always find out something related information within DBpedia datasets which makes it a common target for linking up. DBpedia covers such a big area of information due to its objective to publish structured data from Wikipedia. So in the sense, DBpedia is actually acting as a gateway to connect related information from different datasets with Wikipedia.

2. Wikipedia

We already know that, Wikipedia [3] is one of the biggest and most popular online encyclopedia that we have ever seen and, it is still growing. Currently it consists of approximately 23 million of articles, among them the English Wikipedia itself holds 4,130,559 articles along with 1461 administrators besides 18,124,585 users [4]. These data may be changed at the time of pulication of this paper. A comparative study of the growth between different language editions of Wikipedia can be done from [5]. Most important aspect of Wikipedia is that, it is not only a web based representation of some textual data. It is a huge collection of structured data which is a key resource of the futuristic web i.e. machine readable Semantic Web. Within Wikipedia these structured data are stored mainly using the Infobox templates. Infoboxes are basically collection of certain key properties representing the important aspects as well as the main features of a particular article using that template. A sample use of Infobox OS is shown in the following Fig 1.

```
| name               | নিলাম্বেক | | screenshot | [[হিন্দি: तृक:সংজ্ঞাণীতের যোগাযোগের মাধ্যমের মাত্রক]] | | caption | [[ছন্দো: সুস্মৃতিঃ প্রামাণ্যের মাত্রক]] | | family | [[ইনফোরিমেশনের প্রক্রিয়া]] | | latest_release_version | ১৩০ [[স্ট্রিমের_ছন্দ]] | | latest_release_date | [[ছন্দ_৩১৫]] | | kernel_type | [[ছন্দস্ট্রাটেজিক ব্যবস্থা]] | | license | GNU General Public License | | working_state | উদ্যোক্তা |}
```

Fig. 1. Sample use of Infobox OS (Adapted from http://bn.wikipedia.org/wiki/%E0%A6%9F%E0%A7%87%E0%A6%AE%E0%A6%AA%E0%A7

3. Mediawiki

MediaWiki is a free and open-source software tool specifically developed for Wikipedia. MediaWiki was developed by a student of University of Cologne named Magnus Manske with an aim to replace the old software platform of Wikipedia used at that time called UseModWiki. The first version of MediaWiki software was released in 2003. UseModWiki was written in perl and it stored all the information in the text files, which definitely limited the functionality of a daily growing online encyclopedia like Wikipedia, while MediaWiki is written in PHP like light-weight but high performance and scalable scripting language and besides this MediaWiki uses MySQL database at the back end to store all the information instead of text files which makes it better substitute for the UseModWiki. MediaWiki is rich with its highly performative functionalities and extensibility. It also supports the need of multilingualism since the developers are well aware of the efforts made by the different editors to publish different Internationalized and Localized versions of the main Wikipedia site (English version of course). Currently MediaWiki user interface comes in different languages to make the creation and editing of wiki pages in different languages easier. But having a
functional and efficient nature, still WikiMedia itself considered as difficult to use by a normal end user. With its profoundly tested programs the use of WikiMedia is not only limited to the Wikipedia today. Different governmental non-governmental sites are also using WikiMedia these days to publish their wiki pages online.

4. DBpedia

DBpedia [6] is a highly anticipated project to publish structured data from Wikipedia. We can point out the main objectives of DBpedia project as follows:

- Extraction of structured data from the Wikipedia site (Specifically Wikipedia pages).
- Representation of these structured data in RDF statement so that a RDF graph evolve.
- Use of Ontology and RDF data models as backend of the RDF graph.
- Making accessible this RDF graph through normal browsers.
- Making accessible the RDF graph against a SPARQL endpoint so that any SPARQL query get entertained.
- Answering any complex query against Wikipedia, was not possible prior to the development of the DBpedia project.

The framework used behind this initiative is known as DBpedia Extraction Framework; in short DEF. DEF is responsible for extraction of structured data from the Wikipedia pages. Within Wikipedia most of the structured data is stored using some particular Infobox Template. It is basically a collection of keyword-value pairs. An example Infobox content is already shown in the Fig 1.

5. Motivation

As we already know Wikipedia is a huge collection of information and it is increasing day by day. In the early beginning it was only limited to the English edition. Today it is expanded in several non-English languages. Some of these are non-Latin character based, like Greek, Korean, Bengali etc. Extracting structured data from these versions of Wikipedia is always challenging. Though despite of all the challenges, some of the contributors already developed some non-English versions of the DBpedia. The leading ones in this category are the Greek DBpedia [7] and Korean DBpedia [8]. They all are developed using the i18n Extension of the DBpedia Extraction Framework. Our aim is to develop a pathway for Bengali version of the DBpedia. Bengali Wikipedia is a promising Wikipedia version which holds much important information. And it is increasing day-by-day. So, needs of a Bengali version of DBpedia is justified.

6. DBpedia Extraction Framework

DEF [9] is also known as DBpedia Information Extraction Framework (DIEF). DEF generally divided into two different modules. In DEF every module has a purpose. Normally they are known as Core Module and Dump Extraction module. Core Module holds the main components of the framework while the Dump Extraction Module is used to handle data extraction related issues.

Mercurial, Java Development Kit, Maven, Scala and Virtuoso server (optional) are the main requirements for the setup of DEF. Mercurial [10] is a source control management tool, free to use. It is required to download the DBpedia Extraction Framework form their code hosting site at sourceforge. Since Most of the code of DEF is written in Java and Scala language, both of them is required to be installed before proceed further. Maven is a free and open-source project from Apache group [11]. It is basically a project management tool. Maven is also needed to be installed in advance since it will be use to install and run the DEF. Current version available to install within Ubuntu operating system is version 2 i.e. Maven 2. Another optional component we may require is the Virtuoso server. It is required to host the Extracted data locally for testing purpose.
6.1. DEF Structure

At present DBpedia Extraction Framework is divided into two different modules known as Core Module and Dump Extraction Module. Core Module is responsible for handling the core issues of the framework where as actual extraction is done by Dump Extraction Module. Core Module is also divided into four different components known as Source, WikiParser, Extraction and Destination as shown in the following Fig 2 adapted from the DBpedia site. The Source component represents actual Wiki pages to be transformed into the RDF graph. WikiParser denotes the parser that is used to transform the source pages into Abstract Syntax Type (AST). Extractor represents the mapping from AST to the actual RDF statements. Whereas Destination represents an abstraction of RDF statement’s destination [9]. Besides these several Ontology classes, data parsers and utility classes are also used by the code to enhance the functionality of the framework. To make the Dump Extraction Module working we need to download the Extraction Framework from the online mercurial repository. Mercurial should be installed prior to this operation. Information about Mercurial is available from [http://mercurial.selenic.com/]. Within Ubuntu mercurial is possible to install with the following command,

```bash
$ sudo apt-get install mercurial
```

Next step is to download the DEF framework itself. To do this following commands are used,

```bash
$hg update dump
```

Above two commands ensures the download of the original Extraction framework with its up to date codes. After this the framework has to be installed with the following command from the extraction folder,

```bash
$mvn clean install
```

This will install the framework within the system but before that maven is required to be installed. Present version of maven is Maven 2 and within Ubuntu we can installed it with the following command,

```bash
$ sudo apt-get install maven2
```

After installing the extraction framework we can run it to extract the structured data in RDF format from Wikipedia dump files by running the following command from dump folder within the extraction framework’s main installation folder,

```bash
$mvn scala:run
```

Before the execution of this command takes place, we have to configure the config.default.properties file as per our need and to save it as config.properties. Content of the modified version of the config files is shown in the following table 1 for easy comparison.
In the codes shown in Table 1 the first three lines are very important according to the execution of the Dump Extraction Module. As we can see the configuration file is nothing but a collection of some key-value pairs. The first key is "dumpDir". It provides the actual location where the original Wikipedia dump files will be extracted. This is also the place where extraction module will look for the Wikipedia dump files for extraction purpose. The second key is "outputDir" used to specify location where the actual extracted structured data in RDF format will be stored. The third key is the "updateDumps". It can holds two values, either true or false. If it is set to true then the Extraction framework will download all the Wikipedia dump files at the preferred location automatically otherwise users have to download the necessary dump files manually at the preferred position to proceed further. There are three more parameters shown in the table as follows, "extractors", "extractors.xx" and "languages". Extractor specifies which extractors to be used by the framework initially. There are different extractors available for use. Some of them are "LabelExtractor", "GeoExtractor", "PageLinkExtractor", "WikiPageExtractor" etc. Each extractor is for different type of structured data extraction. The parameter "extractor.xx" implies the language specific extractors to be used where the trailing "xx" must be substituted with the valid language code. In Table 1 we have used "bn" for Bengali language. Naturally the "language" parameter holds the language code "bn" which denotes the language of the Wikipedia dump DEF operating on.

7. DBpedia Internationalization: I18n Extension

Upto this we state about general Dbpedia Extraction Framework. Since our aim is to step forward towards the Bengali version of DBpedia, we need to know about the Internationalization initiatives already taken by the DBpedia Internationalization committee [12]. Basically these committee is for providing the guidance [13] mainly to support more new developers to work on the localized versions of the DBpedia so that structured data can be extracted from the different Wikipedia pages other than those are in English language. This group of developers also works for the i18n extension of the configuration files of the DBpedia Extraction Framework. This configuration files mainly states how a particular extractor will act while extracting structured data from Wikipedia for a particular language. Normally, these configuration files are required to manipulate manually before running the extraction framework. These configuration files are available under the following folders,

```
.../extraction_framework/core/src/main/scala/org/dbpedia/extraction/config
```

Still now any proper tutorials for configuring these configuration files are not available at the time of this writing. So that some snapshot of our changes to these configuration files for the Bengali DBpedia is given in the following Fig 3, Fig 4 and Fig 5.

In the following Fig 6 the actual internationalization version of the DBpedia Extarction Framework used by the Greek team is shown. As we can see in the figure i18n Extraction Manager consists of thirteen different types of extractors and different types of parsers. As stated in [7] for the Greek DBpedia, only five extractors and four parsers
are modified. Basically these are most common and important places to modify the configuration considering the needs of extraction for a particular language. Some modification done for Bengali language is already shown in some of the previous figures (Fig 3, Fig 4). The framework we are using is the pretty same after the Greek DBpedia development team left to make it available for all other languages with its i18n extensions. Since it is very important to know about the framework working behind the scene we shall share some of the information evolved due to modification done by the Greek team to it. The framework that has been used in the Greek DBpedia is the generic one and also usable by the other languages based on non-Latin character. In general the Greek DBpedia project proposed the I18n filters to be pluggable so that they can easily get merged or removed from the framework as per the time’s requirement. In simple words I18n-DIEF is highly configurable according to the concerning language’s requirement. One of the most important aspects of the Greek work is the use of IRIs to represent the interlanguage links among the different language edition of DBpedia. Before the Greek effort takes place two more teams publish their work for two localized versions of the DBpedia, one is German and the other one is Korean DBpedia. The German project used the normal percent coded URI for the interlanguage links while the Koreans used the localized IRIs. More on that they both didn’t use the DBpedia ontology especially coded by the engineers manually considering the most highly
used Infobox Templates. Since both of them didn’t use the DBpedia Ontology, their information extraction scheme proved to be weak. In current version of I18n-DIEF IRI filters are used to represent the resources in UTF-8 character based IRI form. URI is discarded simply because they don’t support the non-Latin characters. URIs represents the non-Latin characters in percent encoded form, not suitable for every condition with non-Latin character based languages. For de-referencing of the IRIs they used the Transparent Content Negotiation (TCN) rules. I18n-DIEF is a combination of different types of extractors, parsers and filters. Each are dedicated for a designated job. For example Interlanguage Link (ILL) extractor is responsible to find out links according to the content over the different language edition of Wikipedia. In terms of importance, most common extractors are Infobox extractors. Currently there are two types of Infobox extractors available - Generic and Mapping based. The main difference between the two is that Generic one extracts all the structured data without any use of DBpedia Ontology, where as the Mapping based Infobox extractor is highly dependent on the DBpedia Ontology. At present all the language edition of DBpedia uses the DBpedia Ontology. That’s why the word related problem or more specifically to say the synonymy problem gets resolved. Besides this another important fact realized by the Internationalization committee is that the need of separate namespace for each language edition. Since, in general every language edition in Wikipedia comes with some articles without any presence of equivalent English translation. So during the extraction process they remain unmatched against the English version, which is against the DBpedia convention for naming of the resources. Regarding this, the namespace will be used by the Bengali DBpedia is “bn.dbpedia.org”.

8. DBpedia as Linked Data

The role of DBpedia is not only limited to extract knowledge or structured data from Wikipedia but, it also publish and interlink them as Linked Data [14][15] following its principle. To make this possible hosting of the RDF dataset is required. The DBpedia community mainly uses the Virtuoso UniversalServer from OpenLink Software. It is a open-source and free to use. Virtuoso is a hybrid tool, a combination of RDF data management tool, Linked Data server, RDBMS, SPARQL endpoint provider, web application provider and many more. It fulfills all the needs of the DBpedia community to publish DBpedia content over the Linked Data web. Within a Ubuntu environment it is installed with the following command,

```
$ sudo aptitude install virtuoso-opensource
```

A password needs to be setup. After the installation it is accessible through the conductor page at http://localhost:8890/conductor. A snapshot of the conductor page is given in the Fig 7. To make it usable for DBpedia content a special
VAD package called "dbpedia_dav.vad" need to install either through the conductor page or by executing following command at isql prompt,

```
isql > vad_install(' < vad_file_location > /dbpedia_dav.vad', 0);
```

A tentative guidance about the installation of Virtuoso server and loading of RDF dataset within it to browse from a normal HTML web browser or to query from a SPARQL endpoint is available at "http://virtuoso.openlinksw.com/dataspace/dav/wiki/Main/VirtRDFInsert".

9. Infobox to DBpedia Ontology Mapping

To perform the extraction of structure data we need to map the Infobox properties to the appropriate DBpedia Ontology [16] properties. Most of the times this job is done manually by the contributors of the DBpedia site members having the editor’s privilege. DBpedia mapping [17] facilities also come with a semi-automatic mapping tool [18] which also support different international languages, so that the editors can use them to ease up the process of mapping. As shown in Fig 8 all the mapping code must be enclosed within {{}}. The first keyword used in the code is TemplateMapping denotes the type of mapping. The next parameter is mapToClass. It holds the name of the Ontology class against whom the Infobox Template will be mapped. Every parameter is separated with "—". The third parameter is called "mappings". This parameters itself consists of several lines of actual mapping enclosed within double curly braces again. The first parameter we see within these inner curly braces is "PropertyMapping". Each "PropertyMapping" itself consists of two parameter called templateProperty and ontologyProperty. The actual property of the template that we want to map is declared against the parameter "templateProperty" while the manually selected matching property from the actual Ontology class that already declared against the parameter "mapToClass" is assigned against the parameter "ontologyProperty".

9.1. Introduction to DBPMap

DBPMap is a Java based tool based on SMASG (Semi-automatic Mapping tool and Automatic Suggestion Generator) framework [19] and at its early age to perform Mapping between DBpedia Ontology and the different Wikipedia template. The Development and implementation is still at the experimental stage. Though our initial aim is to make it useful for Bengali Wikipedia templates, but still the i18n extension is not applied on it. Currently we are experimenting it with the English version only. We are using a twofold approach for it. The first approach is based on
the string based matching between the Wikipedia template parameter and the DBpedia Ontology properties. In the second stage the Wordnet based Semantics oriented matching is in our plan to apply. Finally a complete or partially mapping code will be generated as per the code style mentioned by the DBpedia mapping guidelines. This code then can be submitted at DBpedia’s mapping site. DBPMap not only generate the initial mapping code but it will also generate suggestions for other possible combination of template property and ontology property based on the different matching measurement.

10. Testing Extracted Data

The namespace throughout the process used is http://bn.dbpedia.org. Normally if the extracted data is hosted at the official DBpedia sites then it will be accessed at this point. But since sufficient amount of Infobox to Ontology mapping is still pending for the Bengali DBpedia, the amount of extracted data is very small. All the configuration of i18n extension configuration is also under experimentation. So that we choose free version of Virtuoso Server to host the extracted data locally for testing. A sample snapshot of a Bengali DBpedia page hosted at local Virtuoso server is given in the following Fig 9. Fig 9 represents Bengali Dbpedia page about Linux Operating System.

11. Conclusion

DBpedia publishes structured data from Wikipedia as a machine processable on the Linked Data web, so that other applications/agent utilises these information for their benefit. Initially the DBpedia Extraction Framework was used only for the English version of the Wikipedia pages. But still a lot of information remains untouched within the other non-English versions of the Wikipedia. Same thing is happening with the Bengali version of DBpedia also. In fact at the time of this writing there was no Bengali version DBpedia existed for the Bengali Wikipedia. The main contribution throughout the paper remains to focus on, how the developers may exploit the configuration of the DBpedia Extractors and parsers according to the DBpedia Internationalization Committee’s preferences, so that it becomes possible to develop a Bengali version of DBpedia. Our study shows us, still a thorough research and development is required with the available extractor as well as parsers and other key component so that the extraction procedure for the Bengali DBpedia improves more. Besides this the Bengali Wikipedia is completely underdevelopment, immature and very poorly organized. Naturally it causes severe obstacles during the development process of the Bengali DBpedia. Sometimes it becomes a very tedious job to map the Infobox template parameters used behind the Bengali Wikipedia pages to the DBpedia Ontology. Still there are many pages in Bengali Wikipedia existing without using any of the Infobox templates though their English versions are using one. This type of conflicting and confusing situation
needs to be resolved. So, huge amount of effort required to apply over the Bengali Wikipedia also, so that Extraction framework works well for the Bengali language too.

References