Automatic Multilingual Web Documents Metadata Extraction

Mukhamedshin D.R.,
Kurmanbakiev M.I.,
Gataullin R.R.
Institute of Applied Semiotics of the Academy of Sciences of Tatarstan Republic
Kazan, Russia
Email: damirmuh@gmail.com
Email: write@marat.link
Email: ramil.gata@gmail.com

***Abstract*—This article describes the experience of robot development that crawls multilingual web documents, their language identification and extracting the metadata based on the metadata model of corpus manager of the electronic corpus of Tatar language "Tugan Tel".**

***Keywords*—metadata, data mining, web content mining, information retrieval.**

I. Introduction

Multilingual web documents metadata extraction problem is a topical for the national corpus development. The solution of this problem will allow to extract the semantics of corpus documents. The obtained results will help to reduce the amount of unrecognized word forms in the corpus documents.

Text corpuses which composed of web documents exist for many languages. Some of the most volume are ukWaC for English (about 2 billion word forms), frWaC for French (about 1.6 billion word forms), deWaC for German (about 1.7 billion word forms), itWaC for Italian language (about 2 billion word forms) and others.

Authors was tasked to crawl web documents in the Tatar language and extract metadata of these web documents. This article describes the steps to solve this problem within the context of the Tatar language, but can be applied to other languages.

II. Metadata Representation in the Corpus Manager

According to the EAGLES recommendations set of metadata to describe a text document in an electronic corpus is divided into three parts: external factors, internal factors and technical metadata. In the electronic corpus of the Tatar language "Tugan Tel" set of metadata is also divided into these three parts.

External factors:

- Text type (original or translation);
- Name;
- Author;
- Translator;
- Edition;
- Publishing house;
- Language;
- Creation date;
- The amount of words;
- The amount of words in Russian file;
- Original source;
- Translation source;
- Keywords;
- Copyright information;
- Short description;
- Note.

Internal factors:

- Style;
- Category/theme;
- Place.

Technical metadata:

- Number (ID);
- Source file name;
- Russian file;
- Flag of data validation by moderator.

For automatic web documents crawling an important aspect is the identification of the set of main metadata to be extracted. This set of metadata should be the basis for all received and processed web documents and must properly cover the mandatory (minimum) set of corpus metadata related to external factors. In the case of electronic corpus of the Tatar language "Tugan Tel", these include:

- Name;
- The amount of words;
- Number (ID);
- Source file name.

A. Metadata representation model in the corpus manager

Since the set of documents metadata can contain an unlimited number of properties and be supplemented with the new data in the future, these metadata representation model should ensure the completeness and scalability. The universal solution of the problem faced by the authors, is to use sematic web, which represented in the RDF data representation model. Thus, the information corpus metadata model is a semantic web. As a fundamental model, it was decided to use DCMI recommendations, that describes all of the metadata which are presented in the existing documents of the electronic corpus.

The list of the metadescription properties presented above, allows to define the main objects and relations between them to represent metadata. These are shown in Figure 1 and Table I.

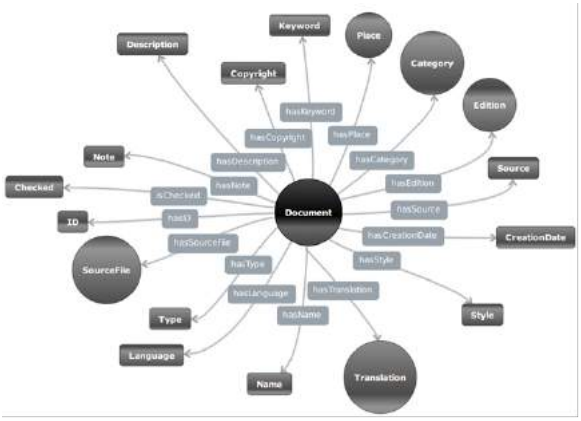

Figure 1. Metadata structure in the corpus manager

Table I. MAIN OBJECTS OF METADATA IN THE CORPUS MANAGER

Properties	Relation	Datatype/Class
Number (ID)	hasID	Integer
Source file, The amount of words	hasSourceFile	SourceFile
Text type	hasType	String
Language	hasLanguage	String
Name	hasName	String
Author	hasAuthor	Author
Translator, Russian file, The amount of words in Russian file, Translation source	hasTranslation	Translation
Style	hasStyle	String
Creation date	hasCreationDate	Date
Original source	hasSource	String
Edition, Publishing house	hasEdition	Edition
Category/theme	hasCategory	Category
Place	hasPlace	Place
Keywords	hasKeyword	String
Copyright information	hasCopyright	String
Short description	hasDescription	String
Note	hasNote	String
Flag of data validation by moderator	isChecked	Boolean

III. THE MAIN METADATA

Automation of the process of crawling web documents requires technical metadata needed for binding objects and identification data obtained.

Web documents crawling module described in this article, use the following technical data:

- Number (ID) of the web document;
- The file name of the web document on the local machine of the module;
- Web document URI in the Internet;
- The file name of the web document metadata on the local machine of the module;
- Data type that stores the web document on the local machine of the module after crawling;
- The amount of data;
- Time saving the web document on the local machine of the module.

These technical metadata stored for each web document and are mandatory for crawling module.

IV. ADDITIONAL METADATA

According to the metadata model in the corpus manager, each document in the system can have a different relation with additional metadata objects. This metadata are usually not semantically marked in web documents, despite the fact that there are recommendations from the W3C for marking those data. This causes some difficulties when trying to extract web document metadata. More details about this will be discussed in Section V.

All the additional metadata are optional, and some web documents can have none of objects of additional metadata.

For the web documents that have been received during the crawling, identified the following additional metadata:

- Language;
- Name;
- Author;
- Translation;
- Creation date;
- Original source;
- Category/theme;
- Place;
- Keywords;
- Copyright information;
- Short description.

V. HOW THE ROBOT WORKS

A. URI collecting for crawling

The first and very important step for the optimization of the robot working is URI collecting for crawling.

To minimize the load on the source machine, it is necessary to optimize requests to the source so that these do not cause problems in the source website working, but given sufficient data for each web document. To do this, the authors tuned URI collecting depending on the characteristics of a source website.

For example, often the quickest and most effective solution was a sequential scan of web pages' identifiers, which does not require additional requests to the web documents with URI list on the source website. This solution, of it is possible to apply to a specific website, not allowed to use the URI list for crawling and performed crawling of web documents directly. Another example is the processing of source website RSS feed, which significantly reduces the number of requests to the source web server. In this case, a single request can obtain data of several web documents, not even request them directly.

B. Web document language identification

Authors was tasked to crawl web documents to compile a corpus of the texts in the Tatar language. In the sources web sites list included as resources exclusively in the Tatar language as multilingual resources, on which the content is located in several languages.

In the case of resources in the Tatar language the language identification is not required. Whereas when crawling multilingual resources necessary to ensure that the text of a web document is written in the Tatar language. In most cases, the web document language can be restricted before crawling, at the stage of URI list collecting. The clear structure of the URI of web document where web document language identifier is often found, contributes to this. Since the robot configuration for each source website is performed manually, this restriction may be applied before the crawling and URI list collecting.

But if it is not possible to restrict the language of a web document content, it is necessary to identify the language and to make sure that the text of a web document is written in the Tatar language. Since the Tatar alphabet includes 6 additional letters (Ә ә, Ө ө, Ү ү, Җ җ, Ң ң, Һ һ), if any in the content of a web document we can confidently say that the language of a web document is Tatar.

C. Crawling and metadata extraction

After web documents URI collecting the robot proceeds to the sequential crawling of these web documents.

Each source website has its own characteristics of web document content representation, so a common approach to all web documents is not applicable. Each website has its own robot configuration file and its own algorithm of processing the content of web documents. The algorithm may consist of any of a finite number of steps, each of which has access to data obtained previously. The use of different types of custom variables, functions for arbitrary data processing, saving data in different formats and structures, stop triggers are defined directly in the robot configuration file and rule changes to the source code of the robot.

One of the most important components of the robot is the variables module. This module allows to crate different types of variables and change them while the robot is working. The variables module supports a variety of mathematical operators, operations with date and time, working with lists, arrays, cache, database, files, and other variables. This module provides sufficient functionality for a complete robot configuration and changing its state during a crawling.

Metadata extraction occurs by means of regular expressions. Depending on the structure of a web document, regular expression can be different in complexity and performed in several steps. An example of a web document that requires processing in several steps is the tatar-inform.ru website web document.

$$\begin{aligned} &< ahref = ".*" = "grey" > (.*) < /a > \\ &< h1 > (.*) < /h1 > .* \\ &< div = "news\_info" > .* \\ &(\backslash d[\hat{}\,,]+), .* < div = "news\_padd" > .* < p \\ &class = "anons" > (.*) < /p > (.*) < /div > \end{aligned} \quad (1)$$

$$\hat{}[\hat{}\ <]* < p > \backslash (([\hat{}\,,]+), .*, ([\hat{}\,,]+)\backslash) \quad (2)$$

The first processing step is shown in (1) and it is a division of the web document structure to the main text, category, name and the block of metadata that require further processing. In the second step (2) from this block are extracted the author's name and location associated with the content of the web document.

The results of the robot working with the tatar-inform.ru website web documents shown in Table II and in the structure (3).

{"name":"Түбән Кама районы аграрийлары көчәйтелгән эш графигына күчә","class":"Авыл"} (3)

Таблица II. Main objects of metadata in the corpus manager

ID	38681
File name	Авыл/http:__tatar-inform_tatar_news_2007_05_11_24643_?print=Y.txt
Type	original
Language	Tatar
Name	Түбән Кама районы аграрийлары көчәйтелгән эш графигына күчә
Author	unknown
Style	not literary
Creation date	11.05.2007
The amount of words	99
Source URI	http://tatar-inform.tatar/news/2007/05/11/24643/?print=Y

VI. Conclusion

The method of automatic web documents metadata extraction proposed in this paper allows not only to automate web documents metadata extraction process, but also as a part of the robot to crawl websites, allows to collect a volume corpus of texts presented on the Internet.

The presented results are part of the data obtained during automatic crawling. Automation of crawling websites in the Tatar language using the robot allowed to obtain the contents of about 200 thousand of web documents, totaling about 57 million word forms.

A common approach in the development of the robot and its components may allow the use of the robot not only with sources in the Tatar language, but also with sources in other languages, without having to change the source code. The architecture of modules provides full functionality for websites crawling and enables the robot to run with any configuration files.

Extracted metadata of web documents will help to further customize the morphological analyzer more accurately, thereby reducing the incidence of morphological ambiguity in the corpus. One of the possible areas of use is also a classification and clustering of texts in the corpus.

Currently we are actively exploring the possibilities of filling the internal factor metadata using semi-automatic or automatic semantic annotation. It is assumed the realization of the possibility of using identification of named entities and assign them to a particular class of the subject entity.

Список литературы

[1] O. Nevzorova, D. Mukhamedshin and M. Kurmanbakiev, Semantic Aspects of Metadata Representation in Corpus Manager System, Open Semantic Technologies for Intelligent Systems (OSTIS-2016), 2016, pp. 371-376.

[2] O. Nevzorova, D. Mukhamedshin and R. Bilalov, Semantic Aspects of Search Request Representation and Precessing in Corpus Manager System, Open Semantic Technologies for Intelligent Systems (OSTIS-2015), 2015, pp. 439-444.

[3] O. Nevzorova, D. Mukhamedshin and R. Bilalov, Corpus Manager for Turkic Languages: Main Functionality, International Conference "Corpus Linguistics - 2015", Saint Petersburg, 2015, pp. 439-444.

[4] D. Suleymanov, O. Nevzorova, A. Gatiatullin, R. Gilmullin and B. Hakimov, National Corpus of the Tatar Language "Tugan Tel": Grammatical Annotation and Implementation, Procedia - Social and Behavioral Sciences, 2013, vol. 95, pp. 68-74.

[5] O. Nevzorova and F. Salimov, Model of Lexicographical Database: Structure, Basic Functionality, Implementation, International Journal "Information Models and Analyses", vol.1, Number 1, 2012, pp. 21-27.

[6] J. McH. Sinclair and J. Ball, EAGLES Preliminary Recommendations on Text Typology EAG—TCWG—TTYP/P, URL: http://www.ilc.cnr.it/EAGLES96/texttyp/texttyp.html, 1996.

[7] D. Hillman, Using Dublin Core, URL: http://dublincore.org/documents/usageguide/, 2005.

[8] Dublin Core Metadata Element Set, Version 1.1, URL: http://dublincore.org/documents/dces/, 2012.

[9] DCMI Metadata Terms, URL: http://dublincore.org/documents/dcmi-terms/, 2012.

АВТОМАТИЧЕСКОЕ ИЗВЛЕЧЕНИЕ МЕТАДАННЫХ МНОГОЯЗЫЧНЫХ ВЕБ-ДОКУМЕНТОВ

Мухамедшин Д.Р., Курманбакиев М.И., Гатауллин Р.Р.

В данной статье рассказывается об опыте разработки робота для обхода многоязычных веб-документов, определении их языка и извлечении метаданных на основе модели метаданных в корпус-менеджере электронного корпуса татарского языка «Туган Тел». В разделе II описывается структура и модель представления метаданных, применяемая в корпус-менеджере. Раздел III раскрывает информацию о необходимых для работы робота технических метаданных. В разделе IV рассказано о дополнительных метаданных, которые могут быть извлечены из веб-документов. V раздел включает в себя описание процесса сбора URI для обхода роботом, метод распознавания языка веб-документа, описание процесса обхода веб-документов и извлечения метаданных.