Titel der Arbeit

Dambowy, Nils

Bachelor Thesis

Institute for Informatics

Goethe University Frankfurt

supervised by:

Dr. Karsten Tolle

Sebastian Grampe

Contents

Statement of authorship

In accord to §25 par. 11 of the examination regulations 2011:

I hereby declare that I am the sole author of this bachelor thesis and that I have not used any sources other than those listed in the bibliography and identified as references. I further declare that I have not submitted this thesis at any other institution in order to obtain a degree.

Frankfurt, the January 13, 2023   Dambowy, Nils

Abbreviations

CN	Corpus Nummorum
RDF	Resource Description Framework
NLP	Natural Language Processing
CN	Corpus Nummorum
CN	Corpus Nummorum
CN	Corpus Nummorum

1 Introduction

Numismatics is the academic discipline focussing on the study of different forms of currency, including e.g. coins, paper money or medals. These objects can be as old as 200 years BC, modern numismatics usually begins at around 1750. Numismatists made it their task to collect, preserve and categorize these archaeological findings and thereby play an important role in the enlightenment of our history. For example when it comes to determining the date of an archaeological site, coins can be of great use since coinage could indicate a time span or the distance from its place of discovery to its origin could tell us about the mobility during these times.

In our current time most numismatic institutions have to go through the challenge of finding an adequate way of managing their data and mostly[1] really on databases. In this thesis I am focussing on the **CN** database, which will be further explained in a later chapter. Currently working with this database is the **D4N⁴** (Data quality for Numismatics based on Natural language processing and Neural Networks)[2] project which goal it is to improve the assignment of existing data(descriptions or images) to different research portals. The assignment is done in two different ways. Firstly the assigning is done with **Natural Language Processing** which extracts conditions and description and lastly image recognition is used to recognize the coin. Furthermore the project aims to continue the development and implementation of tools useful for numismatic research portals.

1.1 Problem description

In this thesis I am redeveloping a currently existing **Natural Language Processing** pipeline, which will be described in more detail in chapter 5.1. The pipeline is consists of two parts. First **Natural Language Processing** is used to extract a name entities (see **Name Entity Relationship**) and word relationships (see **Relationship Extraction**) out of the **CN** database and afterwards written back again. With new data and the programm **D2RQ RDF** data is created with the aim to be published later. Currently the whole process, from the execution of the notebooks to the setup of the D2RQ program has to be done manually which is cumbersome and takes a lot of time.

To improve this, the goal of this thesis is to refine the state of current pipeline by automating the process of the NLP and create RDF data without having to rely on the tools the D2RQ offers. At the end a rest api should handle the execution of the pipeline. When given a coin description as an input it should return the results in the RDF format. This has the upside of having the data in a format that displays it interlinked which makes it easier to see the relationship between different entities.

---

[1]http://nomisma.org/datasets

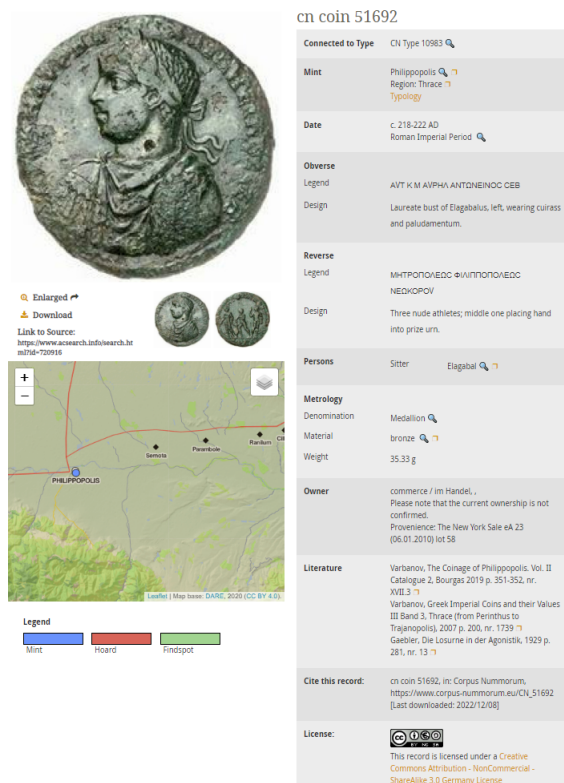[2]http://www.bigdata.uni-frankfurt.de/d4n4/

1.2 Thesis strucutre

In the following chapter I am going go to discuss the CN project and its predecessor, the D4N4 project which is the foundation on which the pipeline is build. In chapter 3 more information about the different technologies used is given to understand the functionality of the new and old pipeline. Afterwards, in chapter 4, the state of current pipeline and the implementation of the revised one is discussed. Both pipelines will also be compared in this chapter. Chapter 5 leaves room for the conclusion and the outlook.

2 Corpus Nummorum and the D4N4

2.1 Corpus Nummorum

The Corpus Nummorum **(CN)** is a research database, which is the result of the joint work of the Münzkabinett Berlin, Berlin-Brandenburg Academy of Sciences and Humanities (BBAW) and the Big Data Lab of Goethe University, with the motivation to offer ancient Greek coinage for research purposes.[3] It contains information about coins with origin in the regions of Lower Moesia, Thrace, Mysia, and the Troad. Added together, the database contains information about approx. 27,500 coins, 14,000 coming from the area of Thrace and 14,500 from the remaining regions. It is also possible to contribute coins to the database yourself.

An example of a coin in the database[2]

All of this information is accessible through the CN Online Website by using the provided search tool, which lets you filter coins by e.g. epoch, tribe, weight or material. Beside the mentioned characteristics, the CN offers text descriptions of the depicted images on the front- and back side of the coins. To allow a better comparison between different coins, all properties have to be entered in accord to a standardized scheme.[3] This not only guarantees the previous mentioned upsides but also make entering coins more accessible since e.g. volunteers do not have to worry about having to come up with a scheme themselves. The scheme offers guidelines on how to describe the obverse and reverse of the coin, figures or architecture, portraits, scenes or some general information on how properly describe a coin. After a submitting a coin, it has to reviewed before it is published.

---

[3]https://www.corpus-nummorum.eu/about

[2]https://www.corpus-nummorum.eu/coins/51692

[3]https://www.corpus-nummorum.eu/pdf/ExternalCoinEntry.pdf

2.2   D4N4

Abbildungen werden immer zentriert, mit einer Bezeichnung versehen, eingefügt.
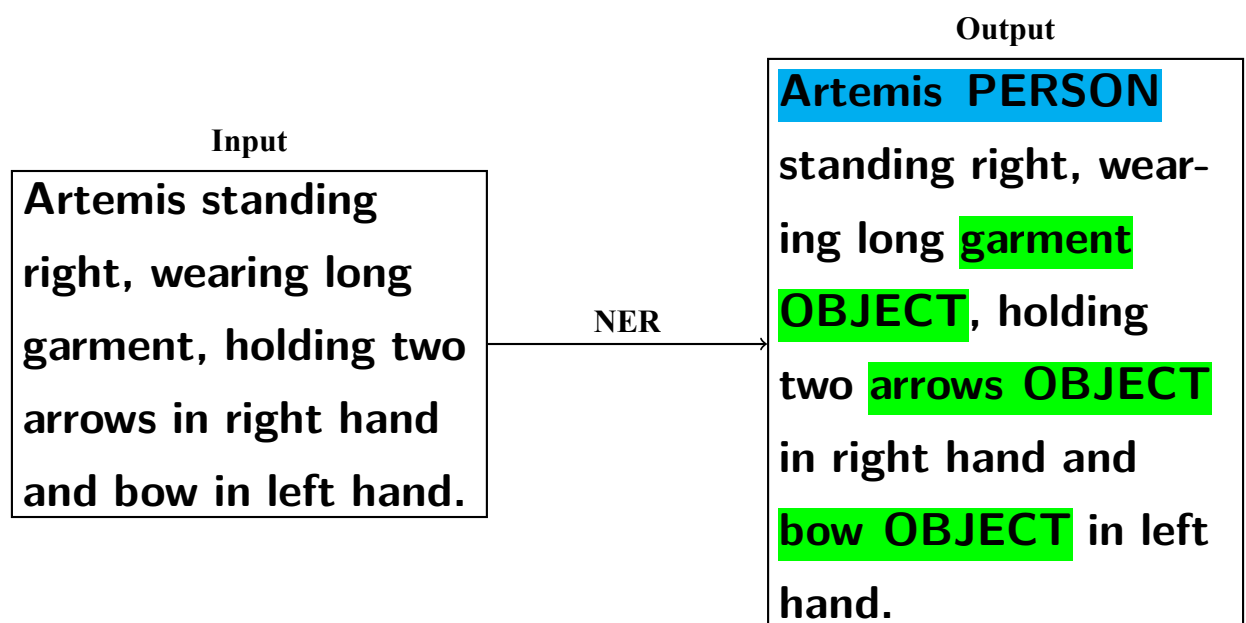
3 Background

3.1 Natural Language Processing

NLP deals with the processing of *natural* Language. The term natural referring to the creation of the language which was natural e.g. from through human conversation unlike artificially created languages like programming languages. The language is given as input in text form for analyzation. Two different forms are being used in the pipeline and are going to be further discussed.w

3.1.1 Named Entity Recognition

In Name Entity Recognition natural text is being analyzed for named entities. Named entities are word which give reference to a real world object using a proper noun. For Example "Goethe" or "Germany", which identify only one object. On the opposite there are words like teacher which does not refer to a specific person. In our case Named Entity Recognition could look like this:

Input

Artemis standing right, wearing long garment, holding two arrows in right hand and bow in left hand.

NER →

Output

Artemis PERSON standing right, wearing long garment OBJECT, holding two arrows OBJECT in right hand and bow OBJECT in left hand.

As shown in the picture *named entity recognition* take a description of a coin and labels the named entities. Altogether there are four different labels, which are being used: **PERSON**, **OBJECT**, **ANIMAL** or **PLANT**.

3.1.2 Relationship Extraction

In text, words are connected over different relationships. Example: Car is a vehicle or car has wheels. With relation ship it is possible to extract this relationship. The relationship is usually between the subject and object of the sentence, thus relationship extraction is focussing on this relation.

Input

("Artemis standing right, wearing long garment,holding two arrows in right hand and bow in left hand.", "Artemis", "garment"),
("Artemis standing right, wearing long garment,holding two arrows in right hand and bow in left hand.", "Artemis", "arrows"),
("Artemis standing right, wearing long garment,holding two arrows in right hand and bow in left hand.", "Artemis", "bow")

Output

("Artemis", "wearing","garment"),
("Artemis","holding", "arrows"),
("Artemis", "holding", "bow")

3.2 Resource Description Framework

A **R**esource **D**escription **F**ramework, often abbreviated as **RDF**, is a standardized model which is most commonly used in the context of the Semantic Web. It is used to describe or exchange graph data. In an **RDF** model, data is represented as a directed graph, which consists out of triple statements.[4]

A triple graph statement is made out of the following components:

- A node for the *subject*

- A node for the *object*

- A *predicate* connecting the two nodes

Fire   wood

S →burns→ **O**
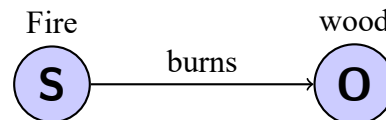
For each of the triple statements exists such graph relationship, and altogether they make up the **RDF** model. Statements inside the **RDF** model that refer to the same subject or object are connected and form a semantic network.

Internally the components of the statement can be represented in different ways. A subject can be either a URI reference or a blank node, an object can be either a URI reference, blank node or a literal and a predicate can exist only as URI reference.[4] A **Uniform Resource Identifier reference**, often shortened to **URI ref**, is a Unicode string that only is made up from characters out of the ASCII Alphabet[5]. It is constructed out of five elements: scheme, authority, path, query and fragment.

Altogether, the generic URI syntax looks like this:

foo://example.com:8042/over/there?name=ferret#nose[6]

This shows the similarity to the more popular **URL**, which is special case of an URI. A **blank node** is a node that is neither a URI ref nor a literal, that's why it is also called an anonymous resource. A **literal** is used to represent values such as numbers, text strings or dates.

---

[4]https://www.w3.org/TR/rdf-concepts/
[5]https://www.ietf.org/rfc/rfc2396.txt
[6]Example taken from: https://en.wikipedia.org/wiki/Uniform_Resource_Identifier

3.3 RDFLib

3.4 D2RQ

To create RDF graphs from the **D4N4** database used in the pipeline, D2RQ is used. D2RQ is a open source mapping tool which allows you to access your data as a RDF graph without having to store it in an RDF format in your database. It consists out of:

- the D2RQ Mapping Language,

- the D2RQ Engine,

- and D2R Server.

However in the current pipeline only the D2RQ Mapping Language and D2R Server are used.

3.4.1 D2R Mapping Language

The D2RQ mapping language is a declarative language used to map the data from the **D4N4** database to RDFS vocabularies. It is responsible for how the virtual RDF graph looks. Example:

```
map:coins a d2rq:ClassMap;
        d2rq:dataStorage map:database;
        d2rq:uriPattern "https://www.corpus-nummorum.eu/coins/@@data_coins.id@@";
        d2rq:class nmo:NumismaticObject;
        d2rq:condition "data_coins.publication_state = 1";
                            .
```

In the example we are creating the 'd2rq:ClassMap' *coins*, which represents one or multiple classes in the D2RQ Mapping Language. Instances of the class are being handled in accord to the definition in 'd2rq:ClassMap' by the D2RQ Mapping Language. In the next line reference to where the data is stored is given by mapping the database to 'd2rq:dataStorage'. In this case database is a variable defined earlier which represents a connection to the database. After this a reference to the resource is given by passing an URI pattern and an RDFS Class is being created. Every instance created from the ClassMap will also be an instance of this class. Lastly we are limiting the mapped coins to those with the value set accordingly, it functions as a SQL WHERE statement.

3.4.2 D2R Server

Ihr Text sollte unter Einbindung von Grafiken und Tabellen in Absätze gegliedert werden. Dabei ist zu beachten, dass ein Absatz einen thematischen Gedanken erfasst, wobei am Anfang des

Absatzes im Regelfall die Kernaussage zu finden ist und von dieser ausgehend durch weitere Erörterungen innerhalb des Absatzes gegliedert wird

4 Assignment and results

4.1 Current state of the pipeline

At the current state the steps of executing the pipeline are associated with a lot of manual work, starting from the manual execution of the NLP notebooks to the running from the D2RQ application.

Summarized the pipeline looks like this:

Test1		Test2		Test3
Description	NER →	**Output**	RE →	**Triples**
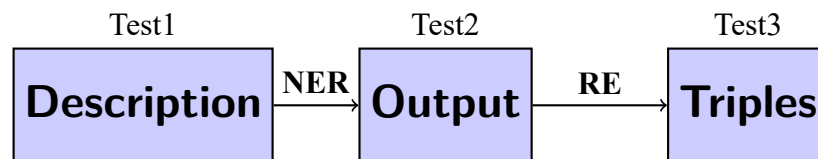
4.2 Implementation of the revised pipeline

4.3 Results

4.4 Comparison to the previous pipeline

5   Conclusion and outlook

6　List of figures

7 Literature

References

Sarkar, Dipanjan (2016). *Text Analytics with Python A Practical Real-World Approach to Gaining Actionable Insights from Your Data*. Apress.