FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Zuzana Šimečková

Entity Relationship Extraction

Institute of Formal and Applied Linguistics

Supervisor of the master thesis: RNDr. Milan Straka, Ph.D.

Study programme: Computer Science

Study branch: IUI

Prague 2020

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In . . . . date . . . .
                                                                              Author's signature

Dedication.

Title: Entity Relationship Extraction

Author: Zuzana Šimečková

Institute: Institute of Formal and Applied Linguistics

Supervisor: RNDr. Milan Straka, Ph.D., Institute of Formal and Applied Linguistics

Abstract: Abstract.

Keywords: key words

Contents

Introduction

This thesis researches relationship extraction in Czech. Relationship extraction is the task of extracting semantic relationship from a text. It is closely connected to named entity recognition, the task of tagging entities in text with their corresponding type, and entity linking, the task of disambiguating named entities to a knowledge base. If all those task are used together, we could gain knowledge databases automatically from text.

For English multiple attempts were made to solve or at least advance in relationship extraction, varying both in task assignment and in used technologies.

To be able to approach this set of tasks, we will focus on pure relationship extraction and thus the following restriction: we will only extract relations from sentences with labeled subject and object for the potential relation. We will benefit from the state-of-the-are technologies such as BERT from Devlin et al. [2018].

A key role in modern machine learning play datasets. In major part of this thesis, we will address the absence of a Czech dataset for relationship extraction. We will generate our dataset by aligning Wikidata[1] with Czech Wikipedia[2]. This type of aligning is sometimes referred to as distant supervision. We will also need to recognize entities includes other . We will than be able to train different models and we will also be able to discuss how choices made in dataset generation affect the ability of a model to learn.

Given the absence of a dataset, we also deal with an absence of a baseline for model performance. To show that, at least the proposed architecture and training method we used, are comparable to state of the art result we will perform the same training with English BERT and we will evaluate it on some well known English datasets.

.

.

0.1 Thesis organization

This thesis is split in two parts. Before we dive into the first part, we will provide information that is relevant for this thesis, but is not part-specific, such as more details on relationship extraction, connected terminology and further motivation. We will briefly introduce the Czech language to explain why existing distant supervision methods were most likely not applied on Czech.

The first part will focus on datasets. We will present some existing supervised datasets, we will propose methodology for generating the dataset via distant supervision and elaborate on the process of implementation and obtained results.

In the second part, we will finally talk about the modern deep learning technologies, we will try to pinpoint the important aspects of models, etc. we are using. We will use the Transformers[3] library which makes training well-known

---

[1]https://www.wikidata.org/wiki/
[2]https://cs.wikipedia.org/wiki/
[3]https://github.com/huggingface/transformers/

Margin notes: divná věta | previous work: Existing work on relation extraction (e.g., Zelenko et al., 2003; Mintz et al., 2009; Adel et al., 2016) | which methods, were they not? | co víc tam je | je tam vizualizovatko

pre-trained models accesible.

whatever
prostě
to
nejdřív
udělej,
pak o
tom piš

1. Relationship extraction intro

1.1 Terminology

Terminology in NLP subtasks is often not exact or non-standardized. We will attempt to introduce following concepts as exactly as possible and respecting the terms that seem to be established by majority.

Relation in this context is semantic (not grammatical etc.). It has a type, is binary and oriented and describes relationship between a relation subject and a relation object.

Relation subject and **relation object**. Subject is the first argument of relation, object the second. In the sentence "Albus Severus is Harry Potters's son." a relation of typeSON is captured, subject is John and object is Eric. The reasoning for this choice of direction is as follows: suppose we are gathering information about Harry, than we would probably have both the information that his son is Albus Severus and his father is James. So we are gathering information about the subject (Harry Potter), even though in most sentences like "James is Harry's father." Harry is a grammatical object. We will use the notation RELATION TYPE(subject,object): SON(Harry Potter,Albus Severus Potter).

Both subject and object can generally be any word or sequence of words that have the ability to form relations. In some cases subjects, objects or both are limited to entities or named entities.

Named entity is a real-world object, such as persons, locations, organizations, products, etc., that can be denoted with a proper name. It can be abstract or have a physical existence. Named entities can simply be viewed as entity instances (e.g., New York City is an instance of a city). Sometimes, numeric data is considered in this category as well (for example by NER tools).

Relation inventory is the set of types of relations, that are considered valid for given dataset or model.

Relation mention is a sentence, that captures a relation, together with type of the relation and tagged subject and object.

Negative mention is close to relation mention in the sense that it is a sentence with tagged subject and object, but the relation type is one of these types:

- OTHER - human annotator would classify a relation of type, that is not in relation inventory.

- NO RELATION - in this case, human annotator should feel an absence of relation between subject and object.

NO RELATION comes with difficulties. Since there is no semantic relation between subject and object, it makes it harder to choose subject-object pairs. It is probably desirable to have subject-object pairs, that could be related in a different sentence.

Relationship Extraction
Lemma

Lexeme
Noun phrase

1.2 Czech language

One of the objective of this thesis is to work with Czech language, therefore we find it useful to make some notes on Czech (for non Czech speaking readers). Czech is a Slavic language with rich morphology and relatively free word order. Most of Czech morphology can be treated with a morphological analyzer, still, it might be useful to have a better understanding of the language we will work with.

1.2.1 Inflection

In Czech, nouns, adjectives, pronounce and numerals are declined. The inflection expresses (not necessarily unambiguously) one of seven cases and a number (singular or plural). Any inflected word in Czech has a grammatical gender, for words, that have natural gender, those two genders align: "žena" (*woman*) is feminime and "muž" (*man*) is masculine. Inflection of each declinable word follows a pattern. This all means that a single word (lemma) can have a lexeme of size

Verbs are conjugated, the conjugation expresses person, numeral, tense, voice and mode. Verbs follow one of 14 patterns and average Czech either finds the theory about Czech verbs and tenses confusing, or is unaware there even are verb patterns. With that, we will not elaborate on conjugation.

An important aspect of declanation for us is agreement. In English, subject and verb agrees (limited just to third person). In Czech subject and verb also agree, but in noun phrases there needs to be an agreement as well.

1.2.2 Free word order

An abstract entity; the set of all forms related by inflection (but not derivation).

příklad: toho, jak je nějaká noun phrase, počet lexémů, počet validních ..

odkaz dopředu, kde řeším, jak matchovat

https://www.aclw 5003.pdf, statistiky o češtině

5

2. Existing datasets

In this chapter, we will overwiev well-known datasets related to Entity Relationship Extraction. We will start with supervised datasets (SEMEVAL 2010 task 8 and TACRED), then we will focus on distant supervision.

2.1 SEMEVAL 2010 task 8 dataset

The SemEval-2010 Task 8 dataset (S10T8) was introduced in SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations Between Pairs of Nominals Hendrickx et al. [2010]. We will summarize how S10T8 was created and some other information from that article so that later we can compare different approaches.

The authors started by choosing an inventory of semantic relations. They aimed for such a set of relations that it would be exhaustive (enable the description of relations between any pair of nominals) and mutually exclusive (given context and a pair of nominals only one relation should be selectable). Chosen relations with descriptions and examples are listed in table 2.1.

They decided to accept as relation arguments any noun phrases with common-noun heads not just named entities or some other specific class of noun phrases, mentioning 'Named entities are a specific category of nominal expressions best dealt with using techniques which do not apply to common nouns.' But they restricted noun phrases to single words with the exception to lexicalized terms (such as science fiction).

The annotation process had three rounds. In the first round, authors manually collected around 1,200 sentences for each relation through pattern-based Web search (with at least a hundred patterns per relation). This way, they obtained around 1200 sentences for each relation. In the second round, each sentence was annotated by two independent annotators. In the third round disagreements were resolved and the dataset was finished. Every sentence was classified either as a true relation mention or was a near-miss and thus classified as "other", or was removed.

The dataset contains of 10717 relation mentions. For the original competition, teams were given three training dataset of sizes 1000 (TD1), 2000 (TD2), 4000 (TD3), and 8000 (TD4). There was a notable gain TD3 →TD4 therefore the authors concluded that even larger dataset might be helpful to increase performance of models. But

> .. that is so much easier said than done: it took the organizers well in excess of 1000 person-hours to pin down the problem, hone the guidelines and relation definitions, construct sufficient amounts of trustworthy training data, and run the task.

2.2 TACRED dataset

The TAC Relation Extraction Dataset was introduced in Zhang et al. [2017]. TACRED is a supervised dataset obtained via crowdsourcing. It contains about

Table 2.1: S10T8 summary. List of relations, their official descriptions, a random example and both relative and absolute count.

CAUSE-EFFECT	12.4%
An event or object leads to an effect.	(1331)
Example: *The <u>burst</u> has been caused by water hammer <u>pressure</u>.*	
INSTRUMENT-AGENCY	6.2%
An agent uses an instrument.	(660)
Example: *The <u>author</u> of a keygen uses a <u>disassembler</u> to look at the raw assembly code.*	
PRODUCT-PRODUCER	8.8%
A producer causes a product to exist.	(948)
Example: *The <u>factory</u>'s products have included flower pots, Finnish rooster-whistles, pans, <u>trays</u>, tea pots, ash trays and air moisturisers.*	
CONTENT-CONTAINER	6.8%
An object is physically stored in a delineated area of space.	(732)
Example: *This cut blue and white striped cotton <u>dress</u> with red bands on the bodice was in a <u>trunk</u> of vintage Barbie clothing.*	
ENTITY-ORIGIN	9.1%
An entity is coming or is derived from an origin (e.g., position or material).	(974)
Example: *The <u>avalanches</u> originated in an extensive <u>mass</u> of rock that had previously been hydrothermally altered in large part to clay.*	
ENTITY-DESTINATION	10.6%
An entity is moving towards a destination.	(1137)
Example: *This book has transported <u>readers</u> into <u>ancient times</u>.*	
COMPONENT-WHOLE	11.7%
An object is a component of a larger whole.	(1253)
Example: *The system as described above has its greatest application in an arrayed <u>configuration</u> of antenna <u>elements</u>.*	
MEMBER-COLLECTION	8.6%
A member forms a nonfunctional part of a collection	(923)
Example: *The <u>student</u> <u>association</u> is the voice of the undergraduate student population of the State University of New York at Buffalo.*	
MESSAGE-TOPIC	8.4%
A message, written or spoken, is about a topic.	(895)
Example: *Cieply's <u>story</u> makes a compelling <u>point</u> about modern-day studio economics.*	
OTHER	17.4%
	(1864)
Example: *The <u>child</u> was carefully wrapped and bound into the <u>cradle</u> by means of a cord.*	

100 000 examples.

The authors are relatively brief about the data collection process:

> We create TACRED based on query entities and annotated system responses in the yearly TAC KBP evaluations. In each year of the TAC KBP evaluation (2009–2015), 100 entities (people or organizations) are given as queries, for which participating systems should find associated relations and object entities. We make use of Mechanical Turk to annotate each sentence in the source corpus that contains one of these query entities. For each sentence, we ask crowd workers to annotate both the subject and object entity spans and the relation types.

TACRED relation inventory captures only relations with subject being an organization or a person. Objects are of following types: cause of death, city, country, criminal charge, date, duration, ideology, location, misc (used for alternative name relation and no_relation only), nationality, number, organization, person, religion, state or province, title and url.

TACRED was designed to be highly unbalanced. 79.5% of data is the no_relation relation, which should be closer to real-world text and supposedly should help with not predicting false positive. However even if we look only at actual relations, there are vast differences in frequency: top six relations make up half the dataset and bottom six less than 2%. In absolute numbers the least common ord:dissolved relation has only 33 examples and median is only 286 examples.

Table 2.2: TACRED summary. List of relations, their official descriptions, a random example and both relative and absolute count.

NO_RELATION Example: " *One step at a time , " said Con Edison spokesman Chris Olert in Sunday editions of The Daily News .*	79.5% (84490)
ORG:ALTERNATE_NAMES Example: *The ARMM was established as a result of the peace agreement between the government and the Moro National Liberation Front -LRB- MNLF -RRB- in 1996 .*	1.3% (1358)
ORG:CITY_OF_HEADQUARTERS Example: *Once completed , the cuts will leave the Irvine , California-based Option One subsidiary with about 1,400 employees .*	0.5% (572)
ORG:COUNTRY_OF_HEADQUARTERS Example: *The Review based its report on a new survey conducted by the International Agency for Research on Cancer in Lyon , France .*	0.7% (752)
ORG:DISSOLVED Example: *News Corp. sold its satellite television service DirecTV in 2008 to Liberty Media .*	0.0% (32)
ORG:FOUNDED Example: *New York-based Zirh was founded in 1995 and makes products using natural oils and extracts .*	0.2% (165)
ORG:FOUNDED_BY Example: *The Jerusalem Foundation , a charity founded by Kollek 40 years ago , said he died of natural causes Tuesday morning .*	0.3% (267)
ORG:MEMBER_OF Example: *Lyons and the Red Sox say they are n't aware of any other Major League Baseball team with such an arrangement .*	0.2% (170)

ORG:MEMBERS	0.3%	
Example: *The NFL refused to abandon the city , and the <u>Saints</u> won the <u>NFC South</u> in 2006 , their first season with Brees and Payton .*	(285)	
ORG:NUMBER_OF_EMPLOYEES/MEMBERS	0.1%	
Example: *Established in September 1969 , the <u>organization</u> now has <u>57</u> member states worldwide .*	(120)	
ORG:PARENTS	0.4%	
Example: *The initial offering of AIA raised $ 178 billion for AIG , while the sale of <u>ALICO</u> to <u>MetLife</u> reaped about $ 155 billion .*	(443)	
ORG:POLITICAL/RELIGIOUS_AFFILIATION	0.1%	
Example: *Manila signed a peace treaty with the <u>MNLF</u> in 1996 , ending a decades-old separatist campaign in return for limited <u>Muslim</u> self-rule .*	(124)	
ORG:SHAREHOLDERS	0.1%	
Example: *Stop the NAACP and <u>Al Sharpton</u> 's <u>National Action Network</u> from committing this disgrace in our community .*	(143)	
ORG:STATEORPROVINCE_OF_HEADQUARTERS	0.3%	
Example: *Learn More <u>Chelsea District Library</u> 221 S Main St Chelsea , <u>MI</u> 48118 -LRB- 734 -RRB- - 475-8732 Find it on a map*	(349)	
ORG:SUBSIDIARIES	0.4%	
Example: *The new law will also enable the government to take over <u>Austral Lineas Aereas</u> , an <u>Aerolineas Argentinas</u> subsidiary .*	(452)	
ORG:TOP_MEMBERS/EMPLOYEES	2.6%	
Example: *Earlier this year , <u>Anatoly Isaikin</u> , head of <u>Rosoboronexport</u> , said Russia still considers Iran a valuable arms customer .*	(2769)	
ORG:WEBSITE	0.2%	
Example: *<u>Swiss Bankers Association</u> : <u>http://www.swissbanking.org</u>*	(222)	
PER:AGE	0.8%	
Example: *Doctor <u>Carolyn Goodman</u> , Rights Champion , Dies at <u>91</u>*	(832)	
PER:ALTERNATE_NAMES	0.1%	
Example: *<u>Remy Ma</u> , whose real name is <u>Remy Smith</u> , is charged with first - degree assault and other charges .*	(152)	
PER:CAUSE_OF_DEATH	0.3%	
Example: *The cause was <u>kidney failure</u> , said a spokesman for the <u>Ali Akbar College of Music</u> .*	(336)	
PER:CHARGES	0.3%	
Example: *Actor <u>Danny Glover</u> has been convicted in Canada for <u>trespassing</u> in a hotel during a union rally in 2006 .*	(279)	
PER:CHILDREN	0.3%	
Example: *<u>Al-Hakim</u> 's son , <u>Ammar al-Hakim</u> , has been groomed for months to take his father 's place .*	(346)	
PER:CITIES_OF_RESIDENCE	0.7%	
Example: *As part of a Navy family , <u>she</u> also lived in Long Beach , Calif. , San Diego and <u>Annapolis</u> .*	(741)	
PER:CITY_OF_BIRTH	0.1%	
Example: *<u>Jane Matilda Bolin</u> was born on April 11 , 1908 , in <u>Poughkeepsie</u> , NY .*	(102)	
PER:CITY_OF_DEATH	0.2%	
Example: *The statement was confirmed by publicist Maureen O'Connor , who said <u>Dio</u> died in <u>Los Angeles</u> .*	(226)	
PER:COUNTRIES_OF_RESIDENCE	0.8%	
Example: *His wife , who accompanied Yoadimnadji to Paris , will repatriate <u>his</u> body to <u>Chad</u> , the ambassador said .*	(818)	

PER:COUNTRY_OF_BIRTH	0.0%
Example: *CARACAS , Jan 10 -LRB- Xinhua -RRB- <u>Hugo Chavez</u> , was born on July 28 , 1954 , in <u>Venezuela</u> 's Sabaneta .*	(52)
PER:COUNTRY_OF_DEATH	0.1%
Example: *Egypt 's state-owned Middle East News Agency said <u>Tantawi</u> died in <u>Saudi Arabia</u> , where he attended a religious ceremony .*	(60)
PER:DATE_OF_BIRTH	0.1%
Example: *<u>Antonioni</u> was born in <u>1912</u> in the northern Italian city of <u>Ferrara</u> .*	(102)
PER:DATE_OF_DEATH	0.4%
Example: *<u>December 6 , 2007</u> <u>Jefferson DeBlanc</u> , Hero Pilot , Dies at 86 By RICHARD GOLDSTEIN*	(393)
PER:EMPLOYEE_OF	2.0%
Example: *<u>He</u> and his group also joined in a legal battle challenging the <u>Washington Redskins</u> ' trademarked name .*	(2162)
PER:ORIGIN	0.6%
Example: *French media are reporting that <u>French</u> tennis player <u>Mathieu Montcourt</u> had died at the age of 24 .*	(666)
PER:OTHER_FAMILY	0.3%
Example: *In the interview <u>Cunningham</u> acknowledged the fragility of <u>his</u> choreographic record .*	(318)
PER:PARENTS	0.3%
Example: *The outgoing governor of Barinas is <u>Hugo de los Reyes Chavez</u> , father of <u>Hugo</u> and Adan Chavez .*	(295)
PER:RELIGION	0.1%
Example: *<u>He</u> closed out the quarter making seven payments to <u>Scientology</u> groups totaling $ 13,500 .*	(152)
PER:SCHOOLS_ATTENDED	0.2%
Example: *<u>She</u> graduated from <u>Mount Holyoke College</u> in 1941 and from the Yale School of Law in 1948 .*	(228)
PER:SIBLINGS	0.2%
Example: *<u>Raul Castro</u> , <u>Fidel</u> 's younger brother , has made several overtures toward Washington .*	(249)
PER:SPOUSE	0.5%
Example: *After returning to Dothan in 1946 , <u>Flowers</u> married <u>Mary Catherine Russell</u> .*	(482)
PER:STATEORPROVINCE_OF_BIRTH	0.1%
Example: *<u>Thomas Joseph Meskill</u> Jr was born in New Britain , <u>Conn</u> , on Jan 30 , 1928 .*	(71)
PER:STATEORPROVINCE_OF_DEATH	0.1%
Example: *Jessica Weiner says <u>Greenwich</u> died of a heart attack at St. Luke 's Roosevelt Hospital in <u>New York</u> .*	(103)
PER:STATEORPROVINCES_OF_RESIDENCE	0.5%
Example: *Sen. <u>Chris Dodd</u> of <u>Connecticut</u> has proposed taxing polluters for their carbon emissions .*	(483)
PER:TITLE	3.6%
Example: *<u>He</u> is the <u>founder</u> and leader of Architects and Engineers for 9/11 Truth -LRB- AE911Truthorg -RRB- .*	(3861)

3. CERED

In this chapter we will describe the process of generating **C**zech **R**elationship **E**xtraction **D**ataset (CERED). We will discuss various decisions that were made during this process and their impacts. We will start by characterizing available data and technological resources.

3.1 Overview

The objective is to create a Relationship Extraction dataset for Czech language using distant supervision. This section is a quick summary for easier orientation in this chapter. Each of these paragraphs is a teaser for one section of this chapter.

First we researched available knowledge bases and Czech corpora to determine which ones will best suit our purpose. We chose Wikimedia projects Wikidata and Czech Wikipedia.

Next we analysed how we will find mentions of Wikidata relations in Czech Wikipedia. We sketched out first dataflow diagrams and thought about all the different complex aspects of this task.

We continued with choosing technologies that we will use. Aware of the volume and other characteristics of chosen data, we chose Python as the main programming language, spark as a way to speed up the computations and MorphoDita to deal with Czech language.

Than we started the implementation and realized that this seemingly simple problem is rather complex. Even though all that we wanted was to get sentences from Wikipedia, find words or phrases, that can have relations, and link those relations from Wikidata, the number of decisions we had to make and obstacles we had to overcome was rather surprising.

As a side project, we implemented a simple viewer, that can present the dataset

3.2 Data sources

To be able to perform distant supervision, we need to find suitable data - Czech text corpus and a knowledge base. We will explain the requirements and constraints we have on such data and present our options. In this section, we will provide more information on the chosen ones.

The main constraint is quite straightforward, there has to be nontrivial shared set of entities and relations mentioned in text and stored in knowledge base. We expect more fact based texts to be more suitable, leaning towards encyclopedic or journalistic genre . One option is to focus on some subset of Czech National Corpus [1], for example SYN2013PUB, SYN2009PUB and SYN2009PUB are corpora of written journalism. The other option is to lean in the direction of encyclopedic text with Czech Wikipedia.

---

[1] https://www.korpus.cz/

Our options for knowledge base are limited, to the best of our knowledge, to Wikidata or Google Knowledge graph [2].

We decided to use Czech Wikipedia and Wikidata, mostly because the intersection of information expressed in text data and in structured data seems promissing.

3.2.1 Czech Wikipedia

Wikipedia is a multilingual online encyclopedia created and maintained as an open collaboration project by a community of volunteersWikipedia contributors [2020] and we believe anyone reading this article is familiar with Wikipedia. From out point of view Wikipedia is a corpus of text with tagged topics of articles and some entity mentions. Czech Wikipedia contains approximately 440 000 articles and ranks top 30 across all the different language editions of Wikipedia. [3]

A dump of Czech Wikipedia is about 1,6GB and 770MB when compressed.

3.2.2 Wikidata

Wikidata is a knowledge base, which acts as central storage of the structured data of Wikipedia and other Wikimedia projects. Just like Wikipedia, this project is freely available and edited by users (and bots). It provides the option to query the database online (for small enough queries), but it is also possible to download the database in standard formats.

The database focuses on **items**, which represent objects, entities, concepts, etc. The first data collected in Wikidata were links to multilingual version of Wikipedia articles on the same topic, the same Wikidata item. Each item was assigned an identifier, prefix Q and unique number, referred to as **QID**. A label together with a description of an item should serve as a human readable identifier. Labels, descriptions a optional aliases are language dependant.

Properties, another big concept of Wikidata, can be thought of as categories of items (mother *P25* implies a category of all mothers) or as relations between items (Ron Weasley *Q173998* has a mother *P25* Molly Weasley *Q3255012*). Each property has its **PID**, an identifier consisting of a prefix P and an unique number, and a data type for a value it can be paired with (such as an item, string, url, number or media file).

Information about any item is recorded in statements. Statement is a key-value pair of an property and a value of prescribed data type. For example, for Ron Weasley *Q173998* there are seven statements about his siblings:

- sibling *P3373* Ginny Weasley *Q187923*,

- sibling *P3373* Fred Weasley *Q13359612*,

- sibling *P3373* George Weasley *Q13359613* and so on

.

---

[2]https://developers.google.com/knowledge-graph

[3]As of March 2020 according to https://en.wikipedia.org/wiki/List_of_Wikipedias

Wikidata project contains over 80 000 000 items, which raises requirements on technological resources, that we will need to work efficiently with such data. Json dump of Wikidata takes 110GB of disk space or 37GB if bzip2 compressed.

3.3   Analysis

The process of creation of CERED is mostly an attempt to execute the first two parts of a pipeline we mention in the first chapter. To the best of our knowledge, there is no suitable entity linking tool for Czech. There are tools for Named Entity Recognition that we could theoretically use to our advantage if we decided to focus on named entity only.

Therefore we need to find a way, to get to similar results as the pipeline would get. We do not expect that our CERED generator will be as powerful as the respective dedicated tools would be. We will not try to create a general entity recognition and linking tools - on the contrary, we will exploit any extra information that chosen Wikimedia projects provide.

There are several aspects that we need to think through.

- Dataflow - we already chosen Wikidata and Czech Wikipedia, but we did not discuss how to connect them and what exactly should be the outcome to be able to proceed to locate mentions.

- Entity Matching - suppose we collected a set of entities that could be mentioned in a piece of text and the text. The process of entity matching attemps to mark parts of the text that mention an entity.

- Wikilink mentions - Wikilinks are (mostly) human labeled entity mentions. jo? Utilizing them is the closes we can get to a supervised dataset without actually supervising the dataset.

- Relation Matching - entities are matched, relations are extracted, can we assume that a sentence with two entities that are related is a mention of this relation?

- Relation Inventory - we generated CERED, but the relation inventory is unnecessarily diverse. Moreover, the dataset is extremely unbalanced - the number of mentions per relation varies. And no "no relation" relation is obtained.

- Result Evaluation

3.3.1   Dataflow

We are starting with two files. One being a Czech Wikipedia dump: it is a collection of articles. Each article has, among other information, its title, id, and text. The other is a Wikidata dump. The simplest way of processing those files would be to process them separately and thus obtaining sentences on one side and relations (a relation type with two items) on the other, see **??**. This approach comes with a clear disadvantage. We would lose any additional information to
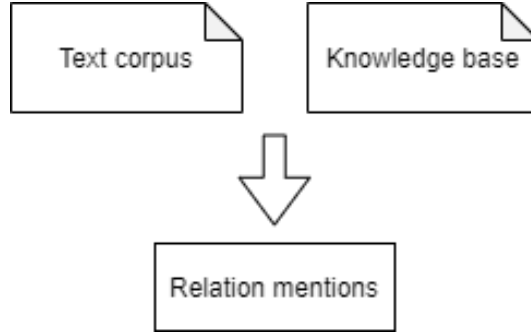
Figure 3.1: Distant supervision diagram

the sentences, that could be potentially useful (for example article title might be helpful to determine which items are mentioned in a sentence). To solve this we could precompute something for each article and attach it to each sentence, risking a massive increase in required capacity to work with such data. On a similar note, if we were to follow the diagram exactly, we would probably store item names (labels and aliases) in each relation, worsening the situation even further.

We decided to update the dataflow to address those issues. We will preprocess Wikidata dump to contain only the data we will use. An item will be kept only if it has a Czech name and we will significantly reduce its statements: we will keep the title of its Czech Wikipedia article and create a list of (QID,PID,QID) triples - **QPQ**, representing statements that contained information about relations between this and other items. This way, we have all the necessary information - article title to be able to connect article to item, names for each item to be able to find mentions of items and finally QPQ triples to connect relations and sentences.

One approach to finding item mentions in text could be called uninformed. We could assume that any item can be mentioned in any sentence. This approach seems to have two main issues: the computation would likely take quite some time but mainly we expect a huge amount of ambiguous mentions. An example of this ambiguity, that we seen as problematic might be children named after their parents. In this case, not only that the entity might get confused, moreover, if we then assign the relation, we might easily confuse a sentence mentioning a spouse relation for a parent relation mention.

On the other side, we can use the extra information that Wikimedia projects provide and opt for a more informed approach. The topic of most Czech Wikipedia articles is a Wikidata item, therefore this item is nearly certainly mentioned in the article. Some Wikidata statements were based on relevant articles and thus it seems rational to expect items, that are related to the main item of an article, to be mentioned. We decided to look only for a tiny subset of all Wikidata items in each article - **candidate items**. As we just discussed, if an article is based on an item, then this item and all items, that are connected to it by a statement, are considered candidate items.

Czech Wikipedia maintains a wbc entity usage table, which contains information about which article uses which item. If we use this table, we are able to obtain a list of items, that should be mentioned in an article, let us call this list a **wbc candidates**. A wbc candidate is at the same time a candidate item.

We might consider adding even a second level of relatives (items related to

14
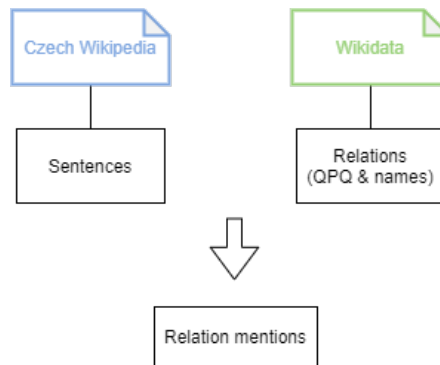
(a) Uninformed approach
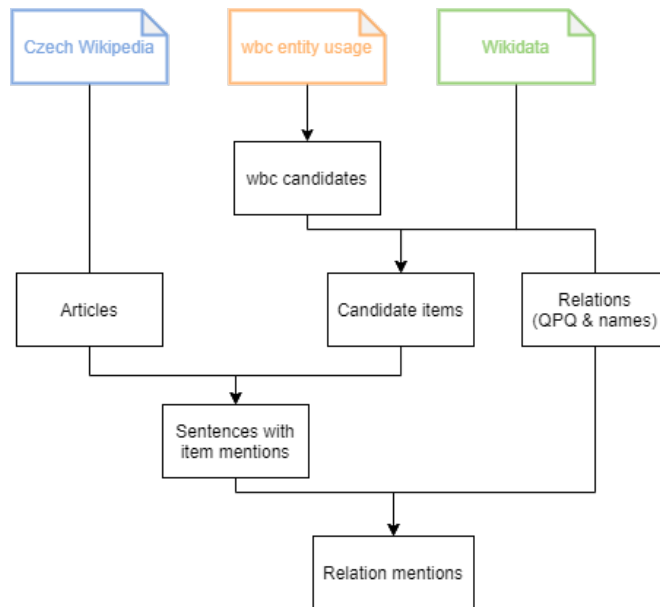
(b) Informed approach

items that are related to the main item) but the branching factor might be relatively high and cause unwanted ambiguity. Consider an instance item like a concrete country, all countries would be second level relatives and thus a candidate item. Since countries tend to be of a certain type (kingdom, republic, state etc.) there might be simply the type or some other more general name amongst their names (Q30 United States of America are also known as America or United States) and more countries might share this name.

So far we mostly discussed the advantages of the proposed informed approach, mainly a hope for higher precision specifically higher precision for item mentions. We should elaborate on some disadvantages as well. We are not trying to fully do entity linking. In the end, we will only use item mentions, if there are two of them in one sentence and if a QPQ that connects them exists. It is debatable whether we need an informed approach to increase relation mention precision, the necessity, and improbability that this condition will be fulfilled for false-positive item mentions might in fact be sufficient.

One more way to locate item mentions is trough **wikilinks**. A wikilink links a page to another page within same-language Wikipedia. First additional information this brings is simply the item mention (if the linked page or article has its main item). We can also consider the textual part of the link to be another name for the linked item. The quality and suitability of this name are to be examined and if we will find these names useful, they can be added to the item names we use.

sice by šlo neparalelně, ale co rychlost?

Mluvit o tom, proč nejdřív najdeme, co v článku hledat, pak to nasekáme na věty, pak matchujeme. Zmínit, kolik je jiných možností, že teoreticky by šlo ještě před rozsekáním na věty dělat entity linking ...

Detailně popsat, co kdy kam poteče + diagram

3.3.2 Entity matching

We have text on one side, gathered candidate items on the other and the goal is to label words that represent those items - entity mentions. There is a wide spectrum of complexity we might aim for, but once again, we are not trying to create a tool for entity recognition and linking. We will describe some of those complexity tears, but before that we should address some basic text manipulation that all of the matching methods will benefit from.

No matter what kind of matching will be done, it seems always beneficial to start with some text preprocessing. Quite a lot of changes need to happen even if some seem like little details. We will separate this preprocessing into a wiki specific part, any text part, and name part.

Wikipedia parsing

Wikipedia is written in Wikitext (Wiki markup, Wikicode). This markup provides all usual functionalities such as determining the layout or fonts and enables commonly-used concepts like lists, links, media file insertion, or tables, and some more wiki specific concepts like infoboxes.

We plan on using some or a combination of existing Wikitext parser since

there seems to be a lot of them. [4] Therefore we believe that the parsing itself will not be troublesome. The more challenging aspect will most likely be deciding, what we consider a valid text for our purpose. From one point of view, a table converted into unstructured text. From one point of view, an unstructured text obtained by converting a table still contains information, that human readers will likely decode. Plus, tables or other structured data tend to contain a lot of information. This will likely cause problems because we want to concentrate on sentence-like data, not just for example tuples of data like a table of athletes might provide (tuples of persons and countries). This kind of data might significantly damage the quality of CERED.

Eliminating all Wikipedia content, that is too structured or generally, not enough sentence-like, but at the same keeping as much as possible will be addressed while implementing, once we can actually see some statistics about such data.

Wikipedia parsing starts with an article in Wikitext and produces human-readable plain text - **clean text**. It should be able to keep track of wikilinks which can be viewed as already human-labeled entity mentions.

Lexical Analysis

Human-readable text is still not necessarily the best for a computer. Splitting character sequences into words and sentences is not as easy as we like to think. A tool that addresses such tasks is usually called a tokenizer. Tokenization is a process that aims to split text (sequence of characters) into separate tokens. Naive tokenizer might just simply split on non-alphanumeric characters. But if it is supposed to actually perform well and recognize sentences a way more sophisticated tool is needed. For this reason, we will outsource handling text to a Czech tokenizer called MorphoDiTa, reasons for choosing this one were discussed in . Using MorphoDiTa tokenization, we can convert clean text into sentences made up of tokens and we even obtain lemma for each token.

Matching methods

Entity matching can be done with various degrees of sophistication. We provide a short overview of those degrees, focusing mainly on those that are to be implemented. For simplicity, we will assume that we are only looking for mentions in one sentence at a time unless written otherwise. We also include an example of such matchings to demonstrate how unsuccessful we are likely to be.

Lord Voldemort (Tom Rojvol Raddle, Voldemort, Pán zla)

Description: First paragraph from Czech Wikipedia page about Lord Voldemort Q176132, also known as Tom Rojvol Raddle, Voldemort, Pán zla (Dark Lord)

String equality. Definitely the easiest method of entity matching. This method is based on a simple substring check which is later extended with additional functionality. In more detail, for each entity, we have multiple name strings and for each of those names, we check whether the name is a substring of the sentence.

---

[4] https://www.mediawiki.org/wiki/Alternative_parsers

We still need to work with letter cases. Named entities should have a fixed letter cases and no additional processing is needed in most cases. In other cases, an established name for a named entity might be written with the lowercased first letter (Weasley family (Q716534) has Czech names 'Weasleyovi' /the Weasleys/ but also rodina Weasleyových /Weasley family/, but there might examples where those different names are completely different (Elizabeth II and Queen of England would be translated to Alžběta II. and královna Anglie). If we consider entities, that are commonly written with lower case (), the sentence now need to be preprocessed so that for example the first letter is not capital. Moreover, there is no guarantee that common names will be lowercased in Wikidata. To conclude, nearly nothing can be assumed about the case of letters, and therefore one of the following solutions might be implemented: everything can be converted to one chosen letter case or some more sophisticated attempts at predicting, which words can have more version in terms of letter cases.

Spaces come with issues as well especially if combined with previous tokenization. We will list some troublesome examples and accept the fact that not everything can be done perfectly. J. K. Rowling has J.K.Rownling as one of Wikidata names, confirming that both versions might appear in written text, but not all entities with similar name type have all space-variants listed in names. We assume that spacing around the '-' character might vary.

It is also not clear if word order in names is fixed for at least for nearly all entity mentions. Even the simple reversion in name, that is sometimes used, will affect the performance of this method (J. K. Rowling and Rowling, J. K.). Even cases where the name is divided by for example apposition into two separate parts might exist.

The greatest weakness of this method is its inability to recognize entities if their name is . To empathize how many words are in Czech text, we colored them in the sample text. We elaborated on Czech language in CHAPTER XX, but just for simplicity - the way verb to be has many different forms (am, are, were, was, would and so on), all nouns and verbs in Czech behave like this, quite often with many more forms.

String similarity (approximate string matching) String similarity still is based on simple string manipulation, no vocabulary or other language knowledge is necessary. The goal is to find entity mention, even if its name is a little altered in the sentence. This alternation can include all of the issues listed for the previous method - cases of letters, spacing, word order, and word forms, but even better, it might help in cases, that we did not anticipate.

There are many metrics describing string similarity. Some could cope better with word order issues, some with word forms, some with spacing. We will not test all of them for our usecase, but still find it useful to mention them since in other than the Czech language, some might work well.

Edit distance First category of string similarity metric is based on edit distance:

Levenshtein distance is the minimum number of edits (additions, deletions, and substitutions of a character) to get from one string to the other. As a metric for example ratio of Levenshtein distance and sum of lengths of strings can be used. This metric, unsurprisingly, deals well with mentions that are close to the names when it comes to the amount of edits needed, so some word forms, different

spacing, and letter casing will be considered a match.

Damerau–Levenshtein distance is very similar to the previous, but a transposition of two adjacent characters is considered an edit. We might argue that some Czech words tend to transpose the last characters in different word forms, but there might be a higher risk of false positives.

One more thing to mention about edit distances is that they count the distance of two string, in our case a name and a substring of a sentence, because we do not expect an entire sentence to be entity mention. Therefore we need to decide on a logic for choosing substrings of a sentence to count the distance on, without diving too deep into this, the time complexity (even though both sentence and name length is relatively small) can be high (Let s, n be lengths of a sentence and a name, there are $s^2/2$ substrings and each edit distance can be computed in sn, leading to s^3n).

Token based Another category is based on tokens. For those metrics, we can either use the tokenized sentences (by an actual tokenizer) and try to tokenize names, so that the format matches, or we can use naive tokenization like removing non-alphanumeric character and splitting on spaces.

The tokenization converts both the sentence string and the name string into a set of tokens (S,N), so a metric that works with sets can be utilized: Intersection over union is computed as —S intersect N— / —S union N— and expresses, or any other set similarity measure can be used.

Token based metrics - due to their set nature - ignore the order of tokens and therefore could solve issues with mentions in which the word order is not the same as in the name. On the other side, an increase in false positives is to be expected and some additional postprocessing is needed to determine which token in the sentence should be considered a mention if it was used in the sentence multiple times. We also feel obligated to mention that once again, we are not trying to find the similarity of a sentence and a name, but of a substring of a sentence and a name. This leads to the idea of removing all tokens, that are in a sentence and would worsen the intersection over union and in result modifying the formula to —S intersect N— / —N—.

Sequence based

Just to be a bit more comprehensive we include another type of metric - sequence based, even though we doubt that it is necessarily the best approach for entity matching. They ignore words as wholes and we do not see any advantage of those metrics for our use case.

Ratcliff-Obershelp similarity finds the longest common substring that is longer than some limit and recursively does so for non-common parts of strings. The result is based on the ratio of ´(double the length of) common parts and overall length.

Bigram (or n-gram) intersection over union which converts both strings into a set of n-grams (n adjacent characters) and performs intersection over union. This time reducing a sentence into a substring is not that straight forward and would require additional attention.

Morphological analysis Moving on from methods that are mostly unaware of the language they work with, we will finally use the tokenized and lemmatized sentence.

Tokenizers (and lemmatizers) are usually trained to perform well on sentences

and might be inaccurate on noun phrases when they stand alone, as entity names do. If we were determined to tokenize them a simple trick like constructing a sentence with this name in it and tokenizing this sentence can partially solve this problem, such sentence that would be grammatically correct and not semantically terrible could be something like "This is /name/", but realistically, this sentence was quite likely not exactly common in the training process of MorphoDiTa. Some foreign words can be erroneous as well and keeping their original form might be the only easy way around it.

With lemmas of both the sentence and the name, we can use any metric from the previous subsection on string similarity (joining the lemmas if a single two strings are needed).

If we decide to keep the names in their original form, we can try to use the correct form of the tokens in the sentence. For each lemma, we get a set of all its possible forms - a lexeme, now we can modify the previous metrics to work with lexemes instead of tokens.

We ended up using a rather simple metric after looking at the results, but we still find it useful to keep this summarization of different string metrics as part of this thesis. A detailed description of the matching method CERED was generated with is written in XX . ⸻⸻⸻⸻⸻⸻⸻⸻ **YY**

zájmena a více vět a tak

A proper entity matching (either in named entity recognition or entity linking) might be expected to recognize entity mention even if the entity is not mentioned explicitly by its name. Pronounces should be assigned an entity they represent (if they do) and other nouns as well. In languages like Czech where the subject of a sentence is often omitted the entity mention is even less obvious but still present. Since the topic of this thesis is not entity matching, we will not debate techniques to achieve this level of matching neither will we implement them.

but the choice of final method was mostly based on performance measured once the implementation was done.

Napsat, že v aj dělají často jen exact modulo zkratky a malé přípony, což tady nejde.

Ukázat nápady se sebráním linků z wikipedie a zavrhnout to

připomenout, jak moc se dá čeština skloňovat říct, že nemá nejspíš smysl snažit se najít jen validní tvary, protože stejně v textu nejspíš nebudou nevalidní

asi mluvit o word order? a možná i implementovat

Říct, že jako kontrolní dataset budou přímo z linků

3.3.3 Wikilink Mentions

As we already mentioned, from our point of view Wikilinks are entity mentions created by Wikipedia editors. The text part of the link can in theory be anything providing us with some more advanced examples of entity linking, that our matching methods cannot perform.

V pátém díle McGonagallová říká, že předmět vyučuje již 39 let

We want to enable the users of CERED to distinguish relation mentions that were created based on two Wikilinks from all others. Since this data is not fully supervised and the word supervised is used often (semi-supervised, distant- **přetížený**

supervised, etc.) we decided to call it **silver**, because they are not of the optimal quality that is usually labeled gold, but they are the best we can get.

3.3.4 Relation Matching

If a sentence contains two entity mentions that are related, chances are that the sentence in fact does express their relationship and thus is a relation mention. This concept is called **distatnt supervision assumption**. The reason why this assumption is used, even though it is clearly not correct, is the how easy it is to use. To be able to tell how often this assumption is violated is labor-intensive, luckily research has been done on this topic. In the **?** the distant supervision assumption is compared to an **express-at-least-once assumption** which states that if two entities participate in a relation, at least one sentence that mentions these two entities might express that relation.

They sampled 600 relation mentions from two corpora, both created by distant supervision on Freebase (knowledge base commonly used before Wikidata took over) and two text corpora - Wikipedia articles and the New York Times corpus. Those 600 samples represented three different relation types (nationality, place of birth, and contains) and were sampled so that there were 100 samples of each type in each corpus. We include their results in table XX They concluded that Wikipedia is a very specific type of text corpora, because articles are centered around entities. We believe that the reasoning can be extended with the fact, that freebase contained information from Wikipedia infoboxes, and those infoboxes were created based on the textual information.

For the authors the results signalized that a more sophisticated tool is needed instead of relying on the distant supervision assumption. We acknowledge that such a tool is needed but at the same time we believe that in our case, where we create CERED based on Wikipedia and Wikidata, the precision they estimated is sufficient. We also assume that Wikidata project is more suitable for this task than was Freebase.

We want to mention that we build CERED to easily fit into the modern deep learning models and be as simple as possible. Therefore the main piece of text we use is a sentence, it might seem intuitive, but it has one downfall. If the relationship extractor was to be used on a real text no to determine where some relations are mentioned but to provide a summary of relations expressed by the text as a whole, some information will be lost.

3.3.5 Relation Inventory

In chapter XX some examples of Relationship classification datasets were introduced. The creators of those datasets claim that in the creation process they first decided on the relation inventory (relation types). Creating the relation inventory seems to be the straight forward and rational approach and we wanted to create such inventory before actually implementing the CEDER generator. We stumbled on those issues.

Wikidata relation inventory (in Wikidata terminology properties) is an order of magnitude larger compared to the traditional relationship extraction datasets and handpicking the inventory is overwhelming. We even considered reducing

the size of this inventory by creating our own relationships that would combine Wikidata properties (parent would be the combination of mother and father relationships). By restricting the Wikidata to just some properties we reduce the size of CERED and if we choose the inventory without considering the number of mentions per relationship, the best-represented relations could be omitted. Knowledge basis, in general, do not contain a relation that could be easily mapped to the "no relation" or "other relation", but for relationship extraction such "mentions" are essential. If we generate mentions using all properties we can later decide which relationships will be in the inventory and the rest of them relabel to "other relation". If we were to assign all tuples of entity mentions that share a sentence and are not related as "no_relationship", we could increase the noise in CERED because not all relations are not in Wikidata and therefore part of the no_relationship mentions could in fact be a positive mention.

While curating the inventory, we should keep in mind, that we are not just choosing the relations but also their mentions and we need to attempt to fulfill those three following requirements to the best of our abilities: Each relation needs to be represented enough. The more balanced relationship representation sizes the better. There should be enough but not too much negative mentions.

3.3.6   Result Evaluation

The by far most challenging aspect of working Czech Wikipedia and Wikidata is their size and diversity. To the best of our knowledge there is no strictly followed guideline when it comes to editing either the articles and item information. Just converting Wikipedia dump to clean text will be challenging due to user defined templates and other constructs we might sill be unaware of. On the other side names in Wikidata can be too general (like someone first name) and create false entity mentions.

We prepared several metrics and methods to measure the quality of the implemented generator and we will list them in the table XX. Unfortunately going through the article and seeing what did and didn't get recognized as a mention is still the most powerful method, therefore we decided to develop a simple streamlit app that can prettify the output of the generator and makes the process of looking at the results nicer.

Quality evaluation

Viewer

The purpose of this viewer is to present the results in a more graphical and pleasant way to fasten and pleasant the process of checking them, which will be done often and will take a significant amount of time.

We want the viewer to be able to show the statistics about the generated data, that were collected during generation and mainly to show an article with the found mentions (entity and relation).

Streamlit made implementing such viewer quite easy and we used spaCy's visualizations.

https://www.aclweb.org/anthology/P19-1074.pdf

jako tabulku, kde purní sloupec je míra automatizace

footnote

screen

do sekce o čeština přidat velká malá pismenka

FA	The overall count of found mentions - to some extent the greater the better - at least in the beginning with naive matching methods. With more sophisticated matching, this number should increase. On the other side, this measure will decrease with more precise Wikitext manipulation.
FA	If the distribution of the amount of found mentions (both the entity mentions before relation matching and the relation mentions) over some domain is peculiar (contains abnormalities such unexpected peeks) the quality of not only those mentions might be lower. The domain can be anything from the following or even multiple of the at the same time: sentence, article, relation, entity, sentence number (the order of the sentence in an article), entity pair.
SA	Check the anomalies
NA	Article Human check -viewer
FA	label by hand and create test set...

Figure 3.2: Quality evaluation methods and the potential for automatization (Fully Automatic, Semi-Automatic, Not-Automatic)
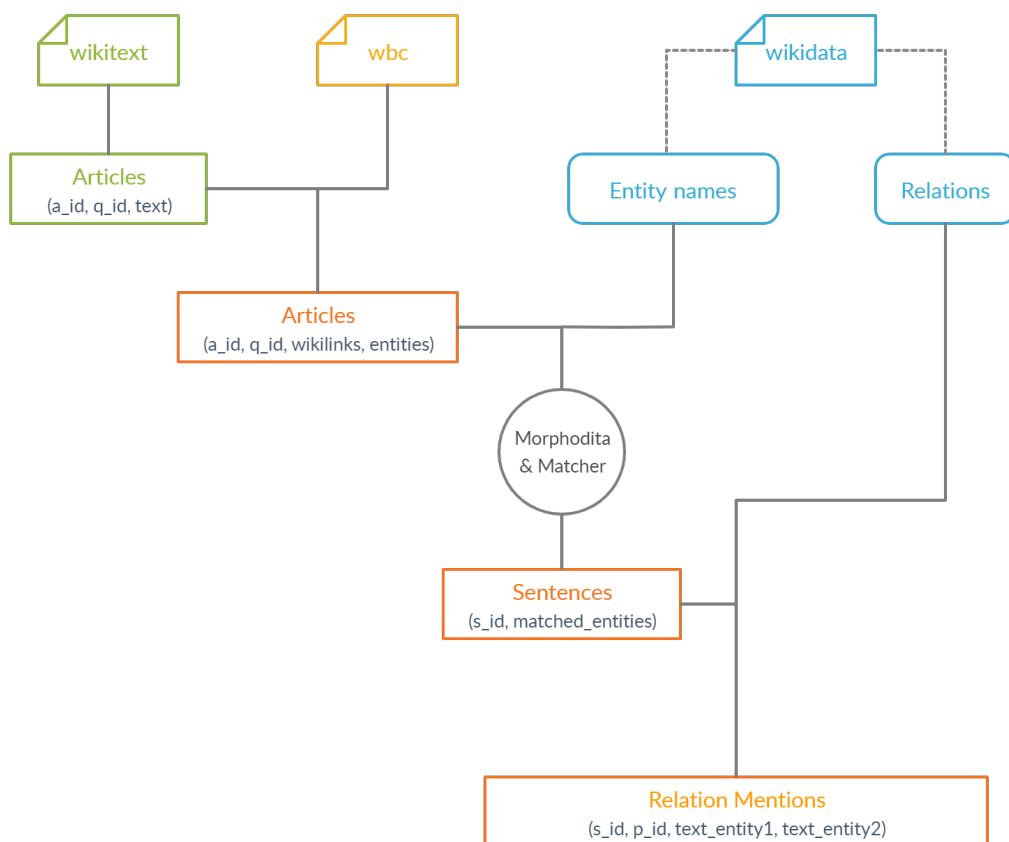

Figure 3.3: Zjednodušený diagram výroby korpusu

3.4   Used technologies

We chose Python to be our main programming language. To be able to work faster with bigger volume of data, we wanted to use a cluster, which leads to Spark and occasionally to some small scripts in shell/bash. To top it, we will use MorphoDita to work with Czech language. Later, we will mention a simple Streamlit app we used to comfortably see results of our Spark queries.

In this section we will briefly introduce those technologies.

3.4.1   Python

3.4.2   Spark

3.4.3   MorphoDiTa

MorphoDiTa Straková et al. [2014] (Morphological Dictionary and Tagger) is an open-source tool for morphological analysis of natural language texts. It is designed to work well on inflective languages and achieves state-of-the-art results for Czech language. Internally, during training tries are built to represent patterns for declension. Externally, MorphoDiTa API provides functionalities such as splitting text into sentences, tokenization and lemmatization.

3.4.4   Streamlit

3.5   Implementation

3.5.1   WikiManipulace

Popsat, že wikitext má sám o sobě strukturovaná a divná data a že je to třeba nejspíš řešit Přesunout do implementace?

3.6   Viewer

3.7   Results

koukli jsme se na diagram a mysleli, že celé ve spark a tak

co je co?

zmiň pandy a numpy a tak

4. Existing ML věci

Conclusion

Bibliography

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/S10-1006`.

Jana Straková, Milan Straka, and Jan Hajič. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Baltimore, Maryland, June 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P14/P14-5003.pdf`.

Wikipedia contributors. Wikipedia — Wikipedia, the free encyclopedia, 2020. URL `https://en.wikipedia.org/w/index.php?title=Wikipedia&oldid=947302871`. [Online; accessed 28-March-2020].

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 35–45, 2017. URL `https://nlp.stanford.edu/pubs/zhang2017tacred.pdf`.

A. Attachments

A.1 First Attachment