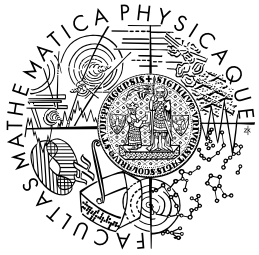


FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

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].

divná věta

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.

.

.

previous work: Existing work on relation extraction (e.g., Zelenko et al., 2003; Mintz et al., 2009; Adel et al., 2016)

0.1 Thesis organization

This thesis is split into 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.

which methods, were they not?

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 on the results we obtained.

co víc tam je

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. Vysvětlíme, jaké metriky se používají a proč a v čem je s nimi problém. Natrénujeme a zkusíme interpretovat výsledky.

[1] https://www.wikidata.org/wiki/

[2] https://cs.wikipedia.org/wiki/

Přesunout We will use the Transformers[3] library which makes training well-known pre-trained models accesible.

whatever prostě to nejdřív udělej, pak o tom piš

[3]https://github.com/huggingface/transformers/

1. Relationship extraction intro

změnit název

uvod do sekce - Zavedeme termíny, připomínáme pipu, další využití, úvod do češtiny

1.1 Terminology

Terminology in NLP subtasks is often not exact or non-standardized. We will attempt to introduce the most important concepts for our work as exactly as possible while respecting the terms that seem to be already established.

hloupá věta

Relation in this context is an abstraction of a semantic relation, for example a father relation. Relation is of a type (father) is binary (between the son and the father) and oriented (the father and the son are not interchangeable), and describes the relationship between a subject (the son) and an object (the father). We will use the term relation as an equivalent for its type and the term relationship for an instance of the relation.

někam zapracovat příklad na větě, jak je nejdřív najití rozštíření a tak všechno

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

obrázek pro příklad v dalším odstavci

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

will we, lepší vzhled

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 Named Entity Recognition tools). An **entity** is a named entity whose proper name is unknown or unimportant but still is an entity instance.

ukradeno z wiki, ocitovat nebo ukrát od jinud

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

definice kruhem

Positive relation mention is a sentence, that captures a relationship: a relation together with a tagged subject and object. We will omit the word positive unless we want to emphasize the fact.

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:

odrážky

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

- NO RELATION - in this case, human annotators should feel an absence of a relationship between a subject and an object.

NO RELATION comes with difficulties. Since there is no semantic relationship 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 objectives 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 feminine and "muž" (*man*) is masculine. The 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. An average Czech either finds the theory about Czech verbs and tenses confusing or is even unaware of the existence of the verb patterns, some verbs tend to be used in a grammatically incorrect forms even in the official language.

An important aspect of declension for us is agreement. In English, subject and verb agree (limited just to the 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

doplnit příklady, včetně vyloženě ne příkladu

znímit entity z názvu

A form from a lexeme chosen by convention (e.g., nominative singular for nouns, infinitive for verbs) to represent that set. Also called the canonical/base/dictiona form. For every form, there is a corresponding lemma

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

kolik?

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

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.

tady představíme exis-tující dataesty

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.

nějak napsat, že nebudu citovat, ale je to hodně vykradené?

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).

proč není table součást od-kazu?

nechat ut tu citaci?

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.

quote better

formát

lepší uvo-zovky

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 burst has been caused by water hammer pressure.*	
INSTRUMENT-AGENCY	6.2%
An agent uses an instrument.	(660)
Example: *The author of a keygen uses a disassembler to look at the raw assembly code.*	
PRODUCT-PRODUCER	8.8%
A producer causes a product to exist.	(948)
Example: *The factory's products have included flower pots, Finnish rooster-whistles, pans, trays, 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 dress with red bands on the bodice was in a trunk 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 avalanches originated in an extensive mass 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 readers into ancient times.*	
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 configuration of antenna elements.*	
MEMBER-COLLECTION	8.6%
A member forms a nonfunctional part of a collection	(923)
Example: *The student association 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 story makes a compelling point about modern-day studio economics.*	
OTHER	17.4%
	(1864)
Example: *The child was carefully wrapped and bound into the cradle 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 Example: *The NFL refused to abandon the city , and the Saints won the NFC South in 2006 , their first season with Brees and Payton .*	0.3% (285)
ORG:NUMBER_OF_EMPLOYEES/MEMBERS Example: *Established in September 1969 , the organization now has 57 member states worldwide .*	0.1% (120)
ORG:PARENTS Example: *The initial offering of AIA raised $ 178 billion for AIG , while the sale of ALICO to MetLife reaped about $ 155 billion .*	0.4% (443)
ORG:POLITICAL/RELIGIOUS_AFFILIATION Example: *Manila signed a peace treaty with the MNLF in 1996 , ending a decades-old separatist campaign in return for limited Muslim self-rule .*	0.1% (124)
ORG:SHAREHOLDERS Example: *Stop the NAACP and Al Sharpton 's National Action Network from committing this disgrace in our community .*	0.1% (143)
ORG:STATEORPROVINCE_OF_HEADQUARTERS Example: *Learn More Chelsea District Library 221 S Main St Chelsea , MI 48118 -LRB- 734 -RRB- - 475-8732 Find it on a map*	0.3% (349)
ORG:SUBSIDIARIES Example: *The new law will also enable the government to take over Austral Lineas Aereas , an Aerolineas Argentinas subsidiary .*	0.4% (452)
ORG:TOP_MEMBERS/EMPLOYEES Example: *Earlier this year , Anatoly Isaikin , head of Rosoboronexport , said Russia still considers Iran a valuable arms customer .*	2.6% (2769)
ORG:WEBSITE Example: *Swiss Bankers Association : http://www.swissbanking.org*	0.2% (222)
PER:AGE Example: *Doctor Carolyn Goodman , Rights Champion , Dies at 91*	0.8% (832)
PER:ALTERNATE_NAMES Example: *Remy Ma , whose real name is Remy Smith , is charged with first - degree assault and other charges .*	0.1% (152)
PER:CAUSE_OF_DEATH Example: *The cause was kidney failure , said a spokesman for the Ali Akbar College of Music .*	0.3% (336)
PER:CHARGES Example: *Actor Danny Glover has been convicted in Canada for trespassing in a hotel during a union rally in 2006 .*	0.3% (279)
PER:CHILDREN Example: *Al-Hakim 's son , Ammar al-Hakim , has been groomed for months to take his father 's place .*	0.3% (346)
PER:CITIES_OF_RESIDENCE Example: *As part of a Navy family , she also lived in Long Beach , Calif. , San Diego and Annapolis .*	0.7% (741)
PER:CITY_OF_BIRTH Example: *Jane Matilda Bolin was born on April 11 , 1908 , in Poughkeepsie , NY .*	0.1% (102)
PER:CITY_OF_DEATH Example: *The statement was confirmed by publicist Maureen O'Connor , who said Dio died in Los Angeles .*	0.2% (226)
PER:COUNTRIES_OF_RESIDENCE Example: *His wife , who accompanied Yoadimnadji to Paris , will repatriate his body to Chad , the ambassador said .*	0.8% (818)

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

3. CERED

In this chapter we will describe our process of generating **C**z**e**ch **R**elationship **E**xtraction **D**ataset (CERED). We will discuss various decisions that we made during this process and their impacts.

nějaká návaznost - říct, že popisuju, že budu spojovat wikipedie

3.1 Overview

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

link zpátky

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

Next we analyze how we will find mentions of Wikidata relations in Czech Wikipedia. We sketch out dataflow diagrams and we think about all the different complex aspects of this task.

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

The amount of questions and options that raised from the analysis gets at least partially answered and decided during implementation. We tested different configurations and went through the data to determine what will work best. As a result, we generated CERED, or more exactly many different versions of CERED in search of the best one to use for training in the second part of this thesis.

Results

results

3.2 Data sources

To be able to perform distant supervision we need to find suitable data - Czech text corpus and a knowledge base (Figure 3.1). In the first subsection, we will explain the requirements and constraints we have on such data and present our options. In the next two subsections, we will provide more information on the chosen ones.

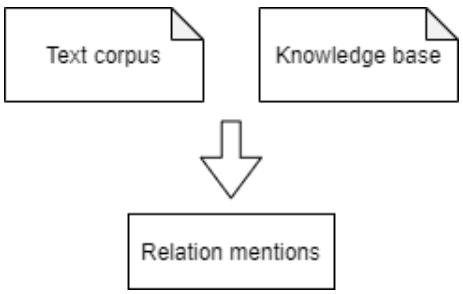


Figure 3.1: Distant supervision diagram

3.2.1 Constraints and requirements

The main constraint is quite straightforward, there has to be a nontrivial shared set of entities and relations mentioned in the text and stored in the knowledge base. We expect more fact-based texts to be more suitable, and therefore we will prefer 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.

tak špatná věta

To the best of our knowledge, our options for knowledge base are limited 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 promising because they are build on each other. Another advantage could be the multilingualism of Wikimedia projects, and therefore the transferability of this work will be higher.

Další důvody.. že jde stáhnout? že není black-box? lepší disambiguita

3.2.2 Czech Wikipedia

Wikipedia is a multilingual online encyclopedia created and maintained as an open collaboration project by a community of volunteers as defined in Wikipedia contributors [2020]. From our 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.

3.2.3 Wikidata

Wikidata is a knowledge base which acts as a central storage of the structured data of 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 a multilingual version of Wikipedia articles on the same topic - on the same Wikidata item. Each item is assigned an identifier, prefix Q and a unique number, referred to as **QID**. A label together with a description of an item should serve as a human readable identifier. Labels, descriptions and 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 a unique number, and a data type for a value it can be paired with (such as an item, string, url, number or media file).

hezčí formátování wiki-itemm

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

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

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

Information about any item is recorded in statements. Statement is a key-value pair of a 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

.

formating

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

jiná slova

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 entities only.

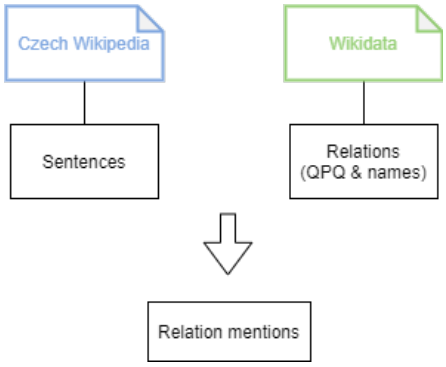
link

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.

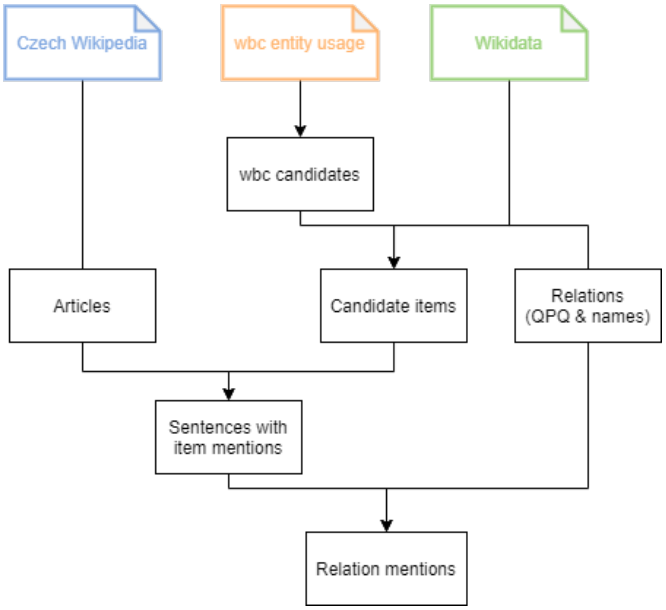
There are several aspects that we need to think through. We introduce them in the following list for better orientation. A subsection in this section is dedicated to each of those aspects.

úplně odstranit ten list?

- Dataflow - we chose 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 locating mentions.
- Entity Matching - suppose we collected a piece of text together wit a set of entities that could be mentioned in the text. The process of entity matching attempts to mark words in the text that mention an entity.
- Wikilink mentions - Wikilinks are (mostly) human labeled entity mentions. Utilizing them is the closest we can get to a supervised dataset without actually supervising the dataset.
- Relation Matching - entities are matched, relationships are extracted, we use distant supervision assumption to locate relation mentions.
- Relation Inventory - we generated CERED, but the relation inventory is overly diverse. Moreover, the dataset is extremely unbalanced - the number of mentions per relation varies. And no “no relation” relation is obtained.
- Result Evaluation - every time we generate CERED during development, we need to evaluate its quality. We propose methods for this evaluation.



(a) Uninformed approach



(b) Informed approach

3.3.1 Dataflow

We start the whole process of creating our dataset with two files. The first file is a Czech Wikipedia dump. It is a collection of articles where each article has its tile, id and text. And the other file 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 relationships (a relation type with two items) on the other. This approach comes with a clear disadvantage. We would lose any additional information to the sentences, that could be potentially useful (for example article title might be helpful to determine which items are mentioned in the sentence).

To solve this we could precompute something for each article and attach it to each sentence (article title, all Wikilinks in the article, etc.) risking a massive increase in required capacity to work with such data. On a similar note, we would probably process Wikidata to store item names (labels and aliases) for each relationship, 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. We will refer to this processed version of Wikidata as **custom Wikidata**. 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 contain information about relations between this and other items. This way, we have all the necessary information - article title to be able to connect an article to an item, names for each item to be able to find mentions of items and finally QPQ triples to connect relations and sentences. Moreover, custom Wikidata size is closer to traditional RAM size. Therefore we could for example load item names into memory, which will come in handy during the implementation.

One approach to finding item mentions in text could be called uninformed (Figure 3.2a). We could assume that any item can be mentioned in any sentence. This approach seems to have two 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, which, unfortunately, is extremely challenging to solve.

On the other side, we can use the extra information that Wikimedia projects provide and opt for a more informed approach. A diagram of this approach is captured in 3.2b 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 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.

typografie

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 the following condition holds: there are two entity mentions in one sentence and 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 suffi-

cient.

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 our goal is to find occurrences of these items in the text. We call this process **entity matching** and each found occurrence is an **entity mention**.

No matter how the matching will be done, it seems always beneficial to start the process 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, lexical analysis and the last part is devoted to lexical analysis on standalone noun phrases.

When we eventually proceed to entity matching, there is a wide spectrum of complexity we might aim for. We are not trying to create a strong sophisticated tool for entity recognition and linking. We will describe some of those complexity tears and choose the right method for our use case.

Wikipedia parsing

Wikipedia parsing starts with an article in Wikitext and produces human-readable plain text - **clean text**. We should keep track of positions of Wikilinks from the Wikitext in the clean text.

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 one or even a combination of existing Wikitext parser since each of them provides different functions.[4] Therefore the parsing itself is not too troublesome.

One problem that needs to be addressed is what should we consider to be a valid text. For example, it is not clear how to work with tables. From one point of view, if we convert a table into an unstructured text, the text will not be a regular text in terms of sentence structure. From a different point of view, an unstructured text obtained by converting a table still contains information, that human readers will likely decode. Moreover, tables and other structured data

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

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

Figure 3.2: First paragraph from Czech Wikipedia page about Lord Voldemort Q176132, also known as Tom Rojvol Raddle, Voldemort, Pán zla (Dark Lord)

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 that for example a table of athletes might provide (tuples of persons and countries). This kind of data might significantly damage the quality of CERED.

The elimination of 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. That way we can see the consequences of the eliminated and kept content.

Lexical Analysis

For our purpose long sequences of characters (such as an article) is not the best format. We need to parse those sequences into smaller tokens, such as words and sentence. 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 the tokenizer is supposed to actually perform well and recognize sentences, a more sophisticated tool is needed. For this reason, we will outsource handling text to a Czech tokenizer called MorphoDiTa, reasons for choosing this specific tokenizer will be discussed in . Using MorphoDiTa tokenization, we can convert clean text into sentences made up of tokens and we even obtain the lemma and lexeme of each token.

link

Lexical Analysis on Names

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 the 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 at all 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.

Matching methods

Entity matching can be done with various degrees of sophistication. We provide a short overview of those degrees (1 – 4), 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 successful we are likely to be.

značka pro překlad v caption

1 String equality. Definitely the easiest method of entity matching. This method is based on a simple substring check which is later extended with ad-

ditional functionality. In more detail, for each entity, we have multiple name variants and for each of those names, we check whether the name is a substring of the sentence.

We still need to work with letter cases. Named entities should have 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 are 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 needs 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 needs to 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.

český

příklady jako škola, tužka

Another problem that we may encounter is how to properly handle spaces. 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 entity names is fixed (or at least almost always fixed). 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 inclined. To emphasize how many words are not in the same form as their lemma in Czech text, we colored them in the sample text. We elaborated on Czech language in CHAPTER XX, but just for simplicity - in English the 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.

vybarvit a dát link

link

2 String similarity (approximate string matching). String similarity is still 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

the ratio of Levenshtein distance and of the sum of the lengths of the 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 mentions differentiating in a word form, different spacing, or letter casing will could 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 and thus this metric could work better for those 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 of a name and of 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 the sentence and the name length is relatively small) can be high if we consider the amount of data we need to process. (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).

vynechat

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 respectively), so metrics that work with sets can be utilized: Intersection over union is computed as $|S \cap N|/|S \cup N|$. 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 the token 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 metric and in result modifying the formula for intersection over union to $|S \cap N|/|N|$.

3 Sequence based. Just to be a bit more comprehensive we include another type of metrics - sequence based, even though we doubt that it is 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 the 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 adjecant 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 morphological analysis we mention earlier.

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 (tokenization on them can be error prone, as we already explained), 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.

4 Advanced concepts.

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.

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 .

mezera

YY

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-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.

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

3.3.4 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** and can be also formulated in the following was: If two entities participate in a relation, all sentences that mention these two entities express that relation. The reason why this assumption is used, even though it is clearly not correct, is how easy it is to use. To tell how often this assumption is violated is labor-intensive, luckily research has been done on this topic. In the Riedel et al. [2010] 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.

table

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 to 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 not 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.

najít článek nebo někam přesunout?

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 the following issues.

Issue 1.Wikidata relation inventory (properties in Wikidata terminology) is an order of magnitude larger compared to the traditional relationship extraction datasets and handpicking our 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 relations).

By restricting the Wikidata inventory 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.

Issue 2.Knowledge basis, in general, do not contain negative relations (such relations that could be easily mapped to the “no relation” or “other relation”), but for relationship extraction negative 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

typografie

typografie

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 relationships are in Wikidata and therefore part of the no_relationship mentions could in fact be a positive mention. The commonly used ratio of negative and positive mentions is .

kolik?

While curating the inventory, we should keep in mind, that we are not just choosing the relations but also their representations and we need to attempt to fulfill the three following requirements to the best of our abilities:

- Each relation needs to be represented enough.
- The more balanced relation representation sizes the better.
- There should be enough negative mentions and their negativity should be assured.

reformulovat

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 or item information. Just converting Wikipedia dump to clean text will be challenging due to user defined templates and other constructs we might 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 3.3. Unfortunately going through the article and seeing what did and did not get recognized as a mention is still the most powerful method, therefore we decided to develop a simple app that can prettify the output of the generator and makes the process of looking at the results less painful.

příliš neformální?

Viewer

The purpose of this viewer is to present the results in a more graphical 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 the generation, the configuration that was used ~~to generate the dataset, 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.~~

footnote

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

přestěhovat někam dopryč tento odstavec

3.4 Used technologies

screen viewertu k implementaci

We chose Python to be our main programming language. To be able to work faster with a bigger volume of data, we wanted to use a cluster, which leads to

do sekce o češtině přidat velká malá písmenka

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 them at the same time: sentence, article, relation, entity, sentence number (the order of the sentence in an article), entity pair.
NA	Checking the anomalies detected by the previous methods.
NA	Checking an article and realizing what did and didn't get matched.
FA	Some articles could be labeled by hand and test could be created.

Figure 3.3: Quality evaluation methods and the potential for automatization (Fully Automatic, Not-Automatic)

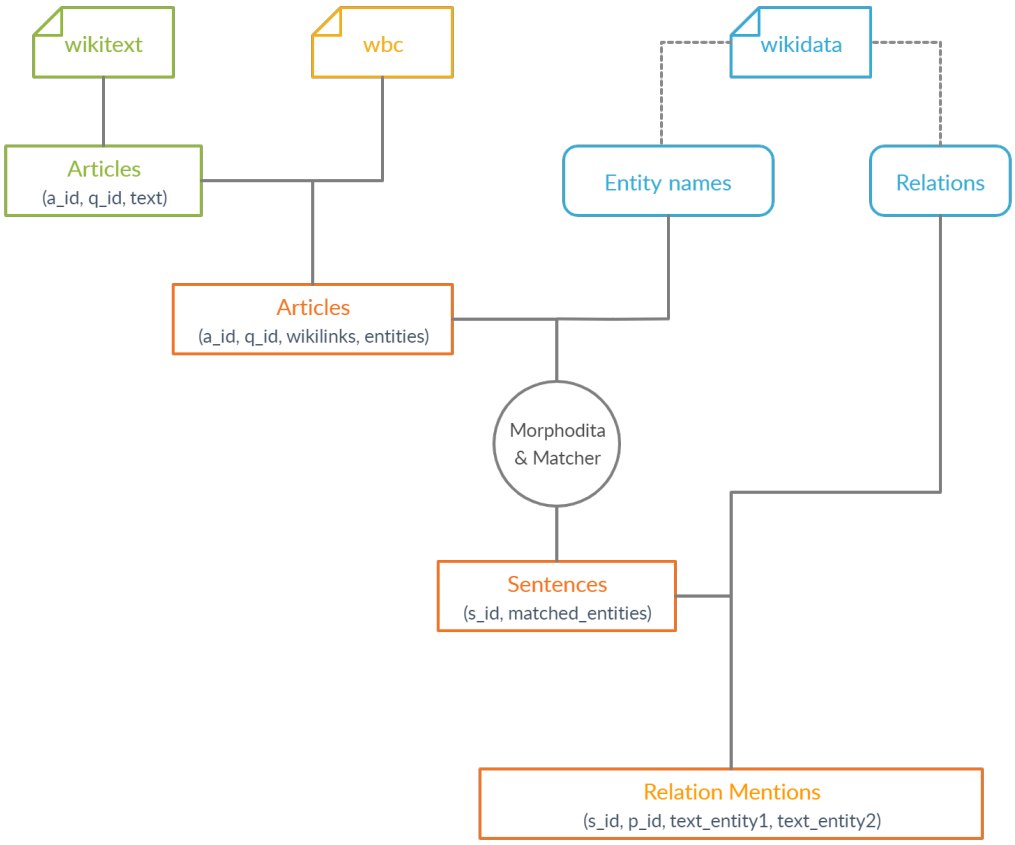


Figure 3.4: Zjednodušený diagram výroby korpusu

Spark and occasionally to some small scripts in shell/bash. To top it, we will use MorphoDiTa to work with the Czech language. Later, we will implement a simple Streamlit app we used to comfortably view the results of our Spark queries.

In this section we will briefly introduce those technologies.

3.4.1 Python

Python je rozšířený, asi nejčastější u AI/ML.

Má spoustu fajn knihoven na všechno (pandy, numpy, scikit, tesorflow, pytorch, transformers ...)

zmiň pandy a numpy a tak

3.4.2 Spark

Umožňuje práci s clusterem

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 the 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

Webb app, zaměřeno na data

3.5 Implementation

In this chapter we answer the questions from 3.3. We elaborate on the details that were not decided and provide more statistic on the used data.

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?

3.5.2 Viewer

3.6 Results

4. Previous Work on Relationship Extraction

In this chapter, we first introduce the most popular pre-trained models that are currently used in NLP. Then we discuss different metrics that are used on Relationship Extraction. Lastly, we research previous work done on the topic of training models for the relationship extraction task.

Pre-trained Models Lately, NLP tasks are dominated by solutions using pre-trained deep neural models. In this section, we introduce BERT, the first well-known model of this type that has set the trend. BERT

Metrics This section focuses on metrics in the Relationship extraction task, we first define those metrics and later discuss the pros and cons of each.

Let us start with metrics for binary classification. In binary classification, we are presented with an input vector and the goal is to determine whether the vector is of class A pro class B. Each prediction then falls into one of the following categories: correctly classified A input, correctly classified B input, wrongly classified A input as B, and a B input wrongly classified as A (Figure 4.1).

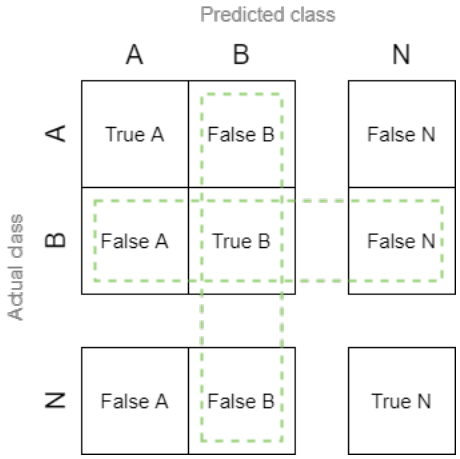


Figure 4.1: Confusion matrix for binary classification

Another way to look at the same situation is to just predict whether an input is of class A or not. Those that way the prediction is a True / False value determining whether the model is of class A. This way, we can define the previously mentioned categories without using the specific classes as **true positive** (TP), **true negative** (TN), **false negative** (FN) and **false positive** (FP). Visualization of the result of classification on a dataset is called a **confusion matrix**, we include such matrix 4.1). We will use the abbreviations to represent the number of predictions that belong to the given category.

without of the number of samples in each category

Accuracy

Accuracy expresses the ration between correct and incorrect predictions.

$Acc = \frac{TP+TN}{FP+FN}$

Precision

Precision expresses the ratio correctly predicted positives within all predicted

positives. Therefore, precision is a good metric if we want to avoid mistakenly classify falses as positives. $Prec = \frac{TP}{TP+FP}$

Recall

Recall is the complementary metric to precision. It expresses the ratio of all positives that were correctly predicted. In other words, it should be used when we need to find the maximum of positives in the data. $Rec = \frac{TP}{TP+FN}$

Let us show three use cases, each of the defined metrics will be the best fit in one case.

Suppose we have a collection of pictures of cats and dogs for adoption. If we were to classify pictures of cats and dogs based on the animal, we would most likely want to maximize the number of correct predictions. Accuracy would aim exactly for that.

If we knew that some adopters suffer from cynophobia (fear of dogs), suddenly the classifier should accommodate the fact by optimizing precision (where a cat is a positive). Note that precision (and recall) in binary classification will return different values if we swap which class is the positive and which is negative.

If the demand for cats extends supply and therefore we have more dogs than cats in the collection, searching for cats could get harder. In such a case, we would want to make sure that all cats are actually classified as cats, and recall would help with that.

To emphasize that the right choice of metric is significant suppose that we have balanced data (both true and false classes are equally represented). If our classifier just predicted that every input is positive, we would obtain the following: 0.5 accuracy, 0.5 precision, and 1 recall. If we were to predict all negatives accuracy and precision would remain 0.5 but recall suddenly drops to 0. If we were to randomly predict the result with even chances for both classes the expected results are 0.5 for all of those metrics. We just described three very different classifiers and the only thing we learned from accuracy and precision was that they were equally bad, without any insight about them. Recall in contrast successfully gave us insight about what the predictions likely are, but evaluated a bad classifier with the highest possible score.

This whole section is in this thesis mostly to remind us that if we want to score well in a given metric, we will likely exploit the metric even if it might actually worsen our classifier. The choice of a metric for a task determines what gets optimized. Later in this section, we will debate such issues in our case, in the relationship extraction task.

F1

Often we might want a trade-off between being as precise as possible and recalling as much as possible. F1 score is a harmonic mean of precision and recall (scaled to range from 0 to 1): $F1 = 2\frac{PrecRec}{Prec+Rec}$ and is quite widely used in competition tasks.

Metrics for multiclass classification

We already run into issues with asymmetry of precision and recall in binary classification (it is dependent on which class is chosen to be the positive one). We can address this by creating metrics per class. In the previous example about binary pet classification, we would get two sets of metrics, each describing the ability of the classifier to recognize given class apart from the rest.

Now we can easily extend this per class approach to multiclass classification.

The formulas will remain exactly the same, only the way we obtain the TP, FP, TN, and FN values is a little different. In a sense nothing changed - if we imagine that the classifier is still binary then the situation is exactly the same. But if we compute those values out of confusion matrix (for class B) than TP is the value on position [B, B], FP is the sum of all in column B without TP, FN the sum of the row B without TP and the sum of cells outside of the Bth row and column are the TN. (Figure 4.2)

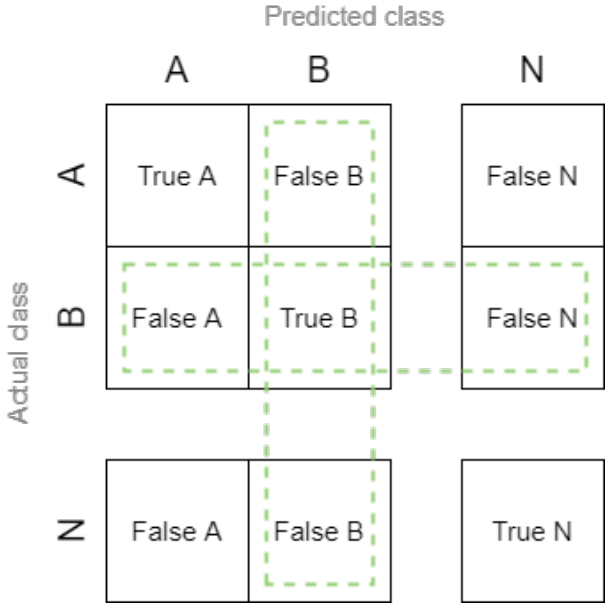


Figure 4.2: Confusion matrix for N-class classification

As a solid way of examining the quality of the classifier, one could simply look at the confusion matrix and at all the per-class metrics. Although this would be insightful, it is not the most practical in terms of a clear comparison of two classifiers. Ideally, we aim for a metric or metrics that are as descriptive and comprehensive as possible but define an ordering of the classifiers.

Weighting metrics

Intuitively we will minimize the number of metrics by combining them into one value. To do so, we should acknowledge that the dataset we evaluate the performance of a classifier and a metric on needs to be taken into consideration.

An ideal dataset would be perfectly balanced. In real life we encounter two types of imbalance in datasets:

class representation distribution (CRD) is not uniform - classes are not equally represented class representation distribution is different in the test and the train part of the dataset

The second imbalance is tricky. Often, when optimizing the classifier, we do not know the CRD of the test dataset. We will therefore mostly focus on the first one.

Macro-averaged metrics

The first method that comes to mind when we aim to combine the same metric of multiple classes into one, is the arithmetic mean. In most libraries and papers the term macro-[metric] (macro-recall, macro-F1, etc.) is used. Macro averaged metrics tend to be the easy option that is used without much thought. So much

so, that even though two macro-F1s are being used, often the exact formula is not included in papers. (The more common formula is the arithmetic mean of classes F1s, but the less often formula where the F1 is computed from macro-recall and macro-precision is also used **?**.)

Weight-averaged metrics If we aimed to pretty much just maximize a metric ignoring the class, instead of averaging the classes with the same weight, we would wight them by their **support**

Whether or not to use macro metrics depends deeply on the use case and the dataset. In our case, the datasets tend to be very imbalanced (the majority of data are negative mentions, and even within the positive mentions the classes are highly unbalanced). This often leads to the idea of computing the combined metric only from positive classes and keeping the metric

Accuracy

FF, TT a spol -¿ precision, F1

Vážení předchozích

Discussion

Relationship Extraction Models

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`.

Sebastian Riedel, Limin Yao, and Andrew Mccallum. Modeling relations and their mentions without labeled text. pages 148–163, 09 2010. doi: 10.1007/978-3-642-15939-8_10.

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