Czech
Technical University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Science

Dataset for Automated Fact Checking in Czech Language

Master Thesis of

Bc. Herbert Ullrich

FCHECK.FEL.CVUT.CZ

Supervisor: **Ing. Jan Drchal, Ph.D.**
Field of study: **Open Informatics**
Subfield: **Artificial Intelligence**
May 2021

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name:	**Ullrich Herbert**	Personal ID number:	**434653**
Faculty / Institute:	**Faculty of Electrical Engineering**		
Department / Institute:	**Department of Computer Science**		
Study program:	**Open Informatics**		
Specialisation:	**Artificial Intelligence**		

II. Master’s thesis details

Master’s thesis title in English:

Dataset for Automated Fact Checking in Czech Language

Master’s thesis title in Czech:

Datová sada pro automatizované ověřování faktů v českém jazyce

Guidelines:

The task is to develop a methodology for collecting a Czech dataset aimed at fact-checking. Collect and process the data and finally perform initial experiments with textual entailment recognition.
1) Explore state-of-the-art in automated fact-checking. Focus on 1) dataset collection methodologies, 2) the last stage of the fact-checking pipelines, i.e., methods of textual entailment recognition.
2) Develop a methodology for Czech fact-checking dataset collection based on the related FEVER [1] Wikipedia-based dataset. The source of data for the new dataset will be the Czech News Agency (ČTK).
3) Create or modify an existing tool and use it to collect dataset.
4) Perform exploratory data analysis.
5) Use the dataset to develop and experiment with initial textual entailment recognition models. These models aim to classify whether textual claims are supported or refuted w.r.t. other reference documents. Evaluate the models using standard approaches used in textual entailment recognition.

Bibliography / sources:

[1] Thorne, James, et al.:FEVER: a large-scale dataset for fact extraction and verification; arXiv preprint arXiv:1803.05355 (2018).
[2] Thorne, James, et al.:The fact extraction and verification (fever) shared task; arXiv preprint arXiv:1811.10971 (2018).
[3] Binau, Julie, and Henri Schulte: Danish Fact Verification: An End-to-End Machine Learning System for Automatic Fact-Checking of Danish Textual Claims; (2020).
[4] Poliak, Adam: A survey on recognizing textual entailment as an NLP evaluation.&quot; arXiv preprint arXiv:2010.03061 (2020).
[5] Storks, Shane, Qiaozi Gao, and Joyce Y. Chai: Recent advances in natural language inference: A survey of benchmarks, resources, and approaches; arXiv preprint arXiv:1904.01172 (2019).

Name and workplace of master’s thesis supervisor:

Ing. Jan Drchal, Ph.D., Department of Theoretical Computer Science, FIT

Name and workplace of second master’s thesis supervisor or consultant:

Date of master’s thesis assignment: **09.02.2021** Deadline for master's thesis submission: ___________

Assignment valid until: **30.09.2022**

Ing. Jan Drchal, Ph.D. Supervisor’s signature	Head of department’s signature	prof. Mgr. Petr Páta, Ph.D. Dean’s signature

III. Assignment receipt

The student acknowledges that the master’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master’s thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt

Student’s signature

Acknowledgements

I would like to thank the 163 students of the Faculty of Social Sciences of Charles University, who donated their time to make our research possible, and to my supervisor, Jan Drchal, for his support, encouragements and proactive participation in every task related to this thesis.

I dedicate this thesis to my girlfriend, Elena Lecce, for supporting me during the hard days before the submission deadline, to Social Bistro Střecha for keeping me nourished throughout the writing process, and to Shia LaBeouf for his ultimate motivational song[1].

Declaration

I hereby declare that I have completed this thesis independently and that I have listed all the literature and publications used. I have no objection to usage of this work in compliance with the act §60 Zákon č. 121/2000Sb. (copyright law), and with the rights connected with the copyright act including the changes in the act.

In Prague, 18. May 2021

[1] `https://youtu.be/UhRXn2NRiWI`

Abstract

Our work examines the existing datasets for the task of automated fact-verification of textual claims and proposes two methods of their acquisition in the low-resource Czech language. It first delivers a large-scale FEVER CS dataset of 127K annotated claims by applying the Machine Translation methods to a dataset available in English. It then designs a set of human-annotation experiments for collecting a novel dataset in Czech, using the ČTK Archive corpus for a knowledge base, and conducts them with a group of 163 students of FSS CUNI, yielding a dataset of 3,295 cross-annotated claims with a 4-way Fleiss' κ-agreement of 0.63. It then proceeds to show the eligibility of the dataset for training the Czech Natural Language Inference models, training an XLM-RoBERTa model scoring 85.5% micro-F_1 in the task of classifying the claim veracity given a textual evidence.

Keywords: Fact-checking, Natural Language Inference, Transformers, BERT

Supervisor: Ing. Jan Drchal, Ph.D.

Abstrakt

Naše práce prozkoumává existující datové sady pro úlohu automatického faktického ověřování textového tvrzení a navrhuje dvě metody jejich získávání v Českém jazyce. Nejprve předkládá rozsáhlý dataset FEVER CS se 127K anotovaných tvrzení pomocí strojového překladu datové sady v angličtině. Poté navrhuje sadu anotačních experimentů pro sběr nativního českého datasetu nad znalostní bází archivu ČTK a provádí ji se skupinou 163 studentů FSV UK, se ziskem 3,295 křížově anotovaných tvrzení s čtyřcestnou Fleissovou κ-shodou 0.63. Dále demonstruje vhodnost datové sady pro trénování modelů pro klasifikaci inference v přirozeném jazyce natrénováním modelu XLM-RoBERTa dosahujícího 85.5% mikro-F_1 přesnosti v úloze klasifikace pravdivosti tvrzení z textového kontextu.

Klíčová slova: Fact-checking, Natural Language Inference, Transformers, BERT

Contents

Part I
Datasets For Fact-Checking in Czech

Part II
Natural Language Inference in Czech

Appendices

Figures

Tables

Chapter 1
Introduction

1.1 Motivation

The spread of misinformation in the online space has a growing influence on the Czech public [STEM, 2021]. It has been shown to influence people's behaviour on the social networks [Lazer et al., 2018] as well as their decisions in elections [Allcott and Gentzkow, 2017], and real-world reasoning, which has shown increasingly harmful during the COVID-19 pandemic [Barua et al., 2020].

The recent advances in artificial intelligence and its related fields, in particular the recommendation algorithms, have contributed to the spread of misinformation on social media [Buchanan and Benson, 2019], as well as they hold a large potential for automation of the false content generation and extraction of sensational attention-drawing headlines – the "clickbait" generation [Shu et al., 2018].

Recent research has shown promising results [Thorne et al., 2019] in false claim detection for data in English, using a trusted knowledge base of true claims (for research purposes typically fixed to the corpus of Wikipedia articles), mimicking the *fact-checking* efforts in journalism.

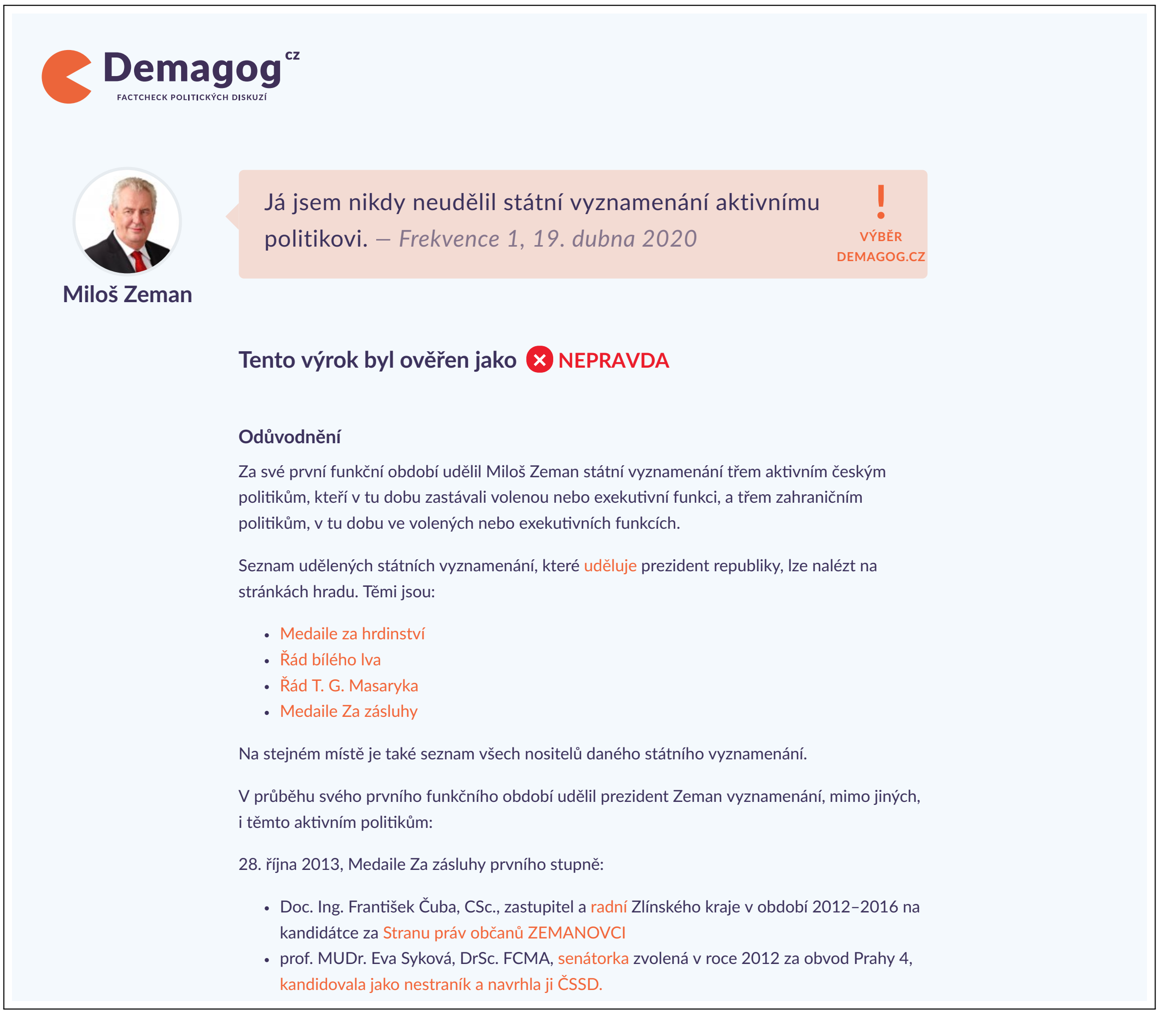
Fact-checking is a rigorous process of matching every information within a *factic claim* to its *evidence* (or *disproof*) in trusted data sources to infer the claim veracity and verifiability. In exchange, if the trusted *knowledge base* contains a set of "ground truths" sufficient to fully infer the original claim or its negation, the claim is labelled as **supported** or **refuted**, respectively. If no such *evidence set* can be found, the claim is marked as **unverifiable**[1].

1.2 Challenges

Despite the existence of end-to-end fact-checking services, such as `politifact.org` or `demagog.cz`, the human-powered approach shows weaknesses in its scalability. By design, the process of finding an exhaustive set of evidence that decides the claim veracity is much slower than that of generating false or misguiding claims. Therefore, efforts have been made to move part of the load to a computer program that can run without supervision.

The common research goal is a fact verification tool that would, given a claim, semantically search provided knowledge base (stored for example as a *corpus* of some natural language), propose a set of evidence (e. g. k semantically nearest paragraphs of the corpus) and suggest the final verdict (Figure 1.2). This would reduce the fact-checker's workload to mere adjustments of the proposed result and correction of mistakes on the computer side.

[1] Hereinafter labelled as `NOT ENOUGH INFO`, in accordance to related research.



Demagog cz

FACTCHECK POLITICKÝCH DISKUZÍ

Miloš Zeman

Já jsem nikdy neudělil státní vyznamenání aktivnímu politikovi. *— Frekvence 1, 19. dubna 2020*

! VÝBĚR DEMAGOG.CZ

Tento výrok byl ověřen jako ✕ NEPRAVDA

Odůvodnění

Za své první funkční období udělil Miloš Zeman státní vyznamenání třem aktivním českým politikům, kteří v tu dobu zastávali volenou nebo exekutivní funkci, a třem zahraničním politikům, v tu dobu ve volených nebo exekutivních funkcích.

Seznam udělených státních vyznamenání, které uděluje prezident republiky, lze nalézt na stránkách hradu. Těmi jsou:

- Medaile za hrdinství
- Řád bílého lva
- Řád T. G. Masaryka
- Medaile Za zásluhy

Na stejném místě je také seznam všech nositelů daného státního vyznamenání.

V průběhu svého prvního funkčního období udělil prezident Zeman vyznamenání, mimo jiných, i těmto aktivním politikům:

28. října 2013, Medaile Za zásluhy prvního stupně:

- Doc. Ing. František Čuba, CSc., zastupitel a radní Zlínského kraje v období 2012–2016 na kandidátce za Stranu práv občanů ZEMANOVCI
- prof. MUDr. Eva Syková, DrSc. FCMA, senátorka zvolená v roce 2012 za obvod Prahy 4, kandidovala jako nestraník a navrhla ji ČSSD.

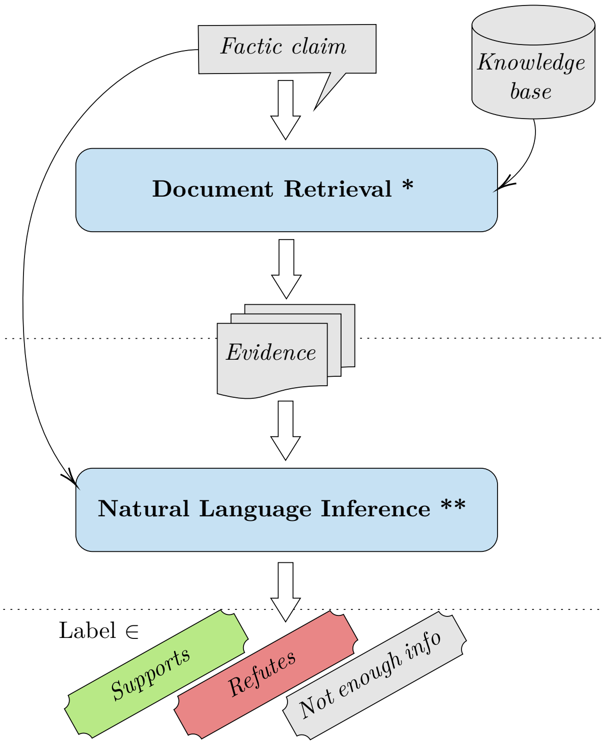
Figure 1.1: Example fact verification from Czech portal Demagog.cz.
Full annotation at `https://demagog.cz/vyrok/19225` – translated in Appendix A.1

The goal of the ongoing efforts of FactCheck team at AIC CTU, as addressed in the works of [Rýpar, 2021,Dědková, 2021] and [Gažo, 2021] is to explore the state-of-the-art methods used for fact verification in other languages, and propose a strong baseline system for such a task in Czech.

1.2.1 Challenge subdivision

In order to maximize our efficiency and the depth of our understanding of every relevant subproblem, we have divided the fact-checking task according to the Figure 1.2 among the members of our research group.

The works of [Rýpar, 2021] and [Dědková, 2021] focus on the Document Retrieval task and compare the performance of the numerical methods, s.a the *tf–idf* search and the *bag-of-words*, to the neural models, most notably the state-of-the-art *Transformer networks* [Vaswani et al., 2017]. [Gažo, 2021] is proposing the methods of their scaling for long inputs, such as full news reports.



* Symbolic (s. a. *tf–idf*-based) or neural search engine that, given a claim c in a natural language, yields a set of paragraphs from the knowledge base thet are semantically closest to c.

To be trained using a dataset matching sample claims to their evidence within the provided knowledge base.

** NLP-based engine that decides whether a provided set of paragraphs entails given factic claim — see **Chapter 7**.

Trained using a dataset mapping evidence sets coupled with claims to their respective labels.

Figure 1.2: Common example of a fact-checking *pipeline* as used in our project

1.2.2 Our contribution

Our part is to provide the needed datasets for the fact verification tasks in the *low-resource* Czech language. We examine both major ways of doing so – localizing the large-scale datasets available in the high-resource languages, typically in English, and collecting a novel dataset through human annotation experiments.

Our second task is to establish a baseline for the final task of the fact-checking pipeline: the *Natural Language Inference*, which is a decisioning problem of assigning a veracity verdict to a claim, given a restricted *set of evidence* in the Czech natural language.

In continuation with research funded by TAČR, experiments are to be made using the archive of the Czech News Agency (hereinafter referred to as ČTK[2]) for a knowledge base, exploring whether a corpus written using journalistic style can be used for such a challenge.

1.3 A word on the Transformers

For the past four years, the state-of-the-art solution for nearly every Natural Language Processing task is based on the concept of *transformer networks* or, simply, *Transformers*. This has been a major breakthrough in the field by [Vaswani et al., 2017], giving birth to the famous models such as Google's BERT [Devlin et al., 2019] and its descendants, or the OpenAI's GPT-3 [Brown et al., 2020].

In our proposed methods, we use Transformers in every step of the fact verification pipeline. Therefore, we would like to introduce this concept to our reader to begin with.

Transformer is a neural model for *sequence-to-sequence* tasks, which, similarly e.g. to the *LSTM-Networks* [Cheng et al., 2016], uses the Encoder–Decoder architecture. Its main

[2] Which stands for "Česká Tisková Agentura", the original name of Czech News Agency

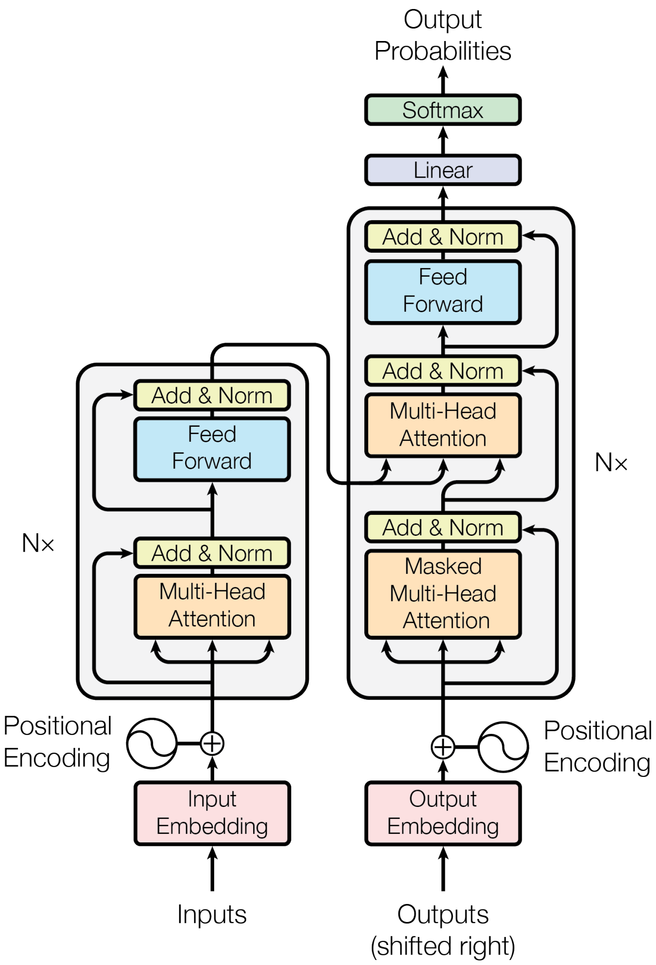


Figure 1.3: Transformer model architecture, reprinted from [Vaswani et al., 2017]

point is that of using solely the *self-attention* mechanism to represent its input and output, instead of any sequence-aligned recurrence [Vaswani et al., 2017].

In essence, the *self-attention* (also known as the *intra-attention*) transforms every input vector to a weighted sum of the vectors in its neighbourhood, weighted by their *relatedness* to the input. One could illustrate this on the *euphony* in music, where every tone of a song relates to all of the precedent ones, to some more than to the others.

The full Transformer architecture is depicted in Figure 1.3.

1.4 Thesis outline

Due to the bipartite nature of our thesis assignment, we have divided the chapters to follow into two parts. The **Part I** presents our Czech datasets and the methods of their collection, and the **Part II** makes the initial experiments for the NLI task.

- **Chapter 1** introduces the problem, motivates the research on the topic and sets up the challenges of this thesis
- **Chapter 2** examines the most relevant research in the field, with an emphasis on the methods of dataset collection, it introduces the two subsequent chapters on the topic

- **Chapter 3** lists and justifies our methods of generating the *localized dataset*, i. e. the methods of transferring the learning examples from a high-resource Natural Language to Czech
- **Chapter 4** describes our methods of collecting a novel fact-checking dataset using the non-encyclopædically structured knowledge base of ČTK news reports
- **Chapter 5** introduces the resulting dataset, as collected during three waves of annotation with Václav Moravec and students of the Faculty of Social Sciences
- **Chapter 6** briefly introduces the full fact-checking pipeline we have established with the FactCheck team at AIC using the collected data and a couple of real-world applications stemming from it
- **Chapter 7** explores the state-of-the-art methods of Natural Language Inference and their potential for our system, and it proceeds to make preliminary experiments on our dataset using these methods
- Finally, **Chapter 8** concludes the thesis, summarises the results we have achieved and proposes directions for future research

Part I

Datasets For Fact-Checking in Czech

Chapter 2
Data Collection

In Chapter 1, we have introduced the framework of automated fact-checking. In order to construct an automated fact verifier, we first need methods of collecting the samples of Czech textual claims and their respective annotations within a fixed knowledge base.

These will allow us to assess the strength of the fact verifier in terms of compliance with human output for the same task. Furthermore, a dataset of sufficient size could be used to train statistical models.

2.1 Related work

As of May 2021, we have reviewed the following most notable papers and projects in the field, so as to provide proofs of concept and strong baselines..

- **Demagog dataset** [Přibáň et al., 2019] – dataset of verified textual claims in low-resource Slavic languages (9082 in Czech, 2835 in Polish, 12554 in Slovak), including their metadata s. a. the speaker's name and political affiliation.

  We have reviewed the Demagog dataset and deemed it not suitable for our purposes, as it does not operate under an enclosed knowledge base and rather justifies the veracity labeling through justification in natural language, often providing links from social networks, government operated webpages, etc.

  Even though the metadata could be used for statistical analyses, the loose structure of the data does not allow its straightforward use for the purpose of training/evaluation of NLP models.

- **FEVER dataset** [Thorne et al., 2018a] – "a large-scale dataset for Fact Extraction and VERification" – dataset of 185,445 claims and their veracity labels from {`SUPPORTS`, `REFUTES`, `NOT ENOUGH INFO`}. Each label (except `NEI`s) is accompanied by a set of all[1] minimum evidence sets that can be used to infer the labelling.

  It was extracted by 50 human annotators from approximately 50,000 popular Wikipedia article abstracts[2] and fact-verified against every abstract in the full June 2017 Wikipedia dump.

  This is the most commonly used dataset used for validation of fact verification pipelines to date, and has been used as a benchmark in shared tasks [Thorne et al., 2018b,

[1] While this is assumed to be true by the FEVER benchmark, there are, in fact, valid evidence sets missing, due to the time constraints for the annotation task. In [Thorne et al., 2018a], 1% annotations were re-annotated by *super-annotators* tasked to find every possible evidence set without a time constraint, which has shown the precision/recall of the regular annotations to be 95.42% and 72.36%, respectively.

[2] The introductory section (i. e. the first paragraph) of Wikipedia article, one before the table of contents.

Thorne and Vlachos, 2019] that inspired the publication of number of well-performing verifiers of English claims [Malon, 2018, Hanselowski et al., 2018, Nie et al., 2019a].

It was collected using a Flask app called the FEVER Annotations Platform, which has been partly open-sourced[3] and thoroughly described in [Thorne et al., 2018a].

- **Danish fact verification datasets** [Binau and Schulte, 2020] – an effort to build an end-to-end fact verifier for the low-resource language of Danish, using the strategies employed by the submissions of the FEVER shared task [Thorne et al., 2018b] and multilingual BERT [Devlin et al., 2019] for the Document Retrieval task.

  Binau and Schulte have handcrafted a dataset of 3,395 textual claims and their labels, along with evidence from the Danish Wikipedia, publishing an open source Command-line interface[4] for this task.

  They have also localized the large-scale FEVER dataset to Danish using the Microsoft Translator Text API and concluded separate experiments on the translated FEVER DA dataset.

We have not found an appropriate dataset for the NLP tasks we pursue, which is a common problem of a the non-international languages, such as Czech. We say that Czech is a *low-resource* language, which, in NLP, signifies the need of adopting the methods and – where possible – the local versions of the corpora used for the tasks on foreign languages.

In order to train a verifier of our own for Czech (and for a whole different domain of the ČTK journal), we have attempted to repurpose the existing annotations of the FEVER dataset, as well as the annotation practices of both [Thorne et al., 2018a] and [Binau and Schulte, 2020] where applicable.

The subsequent chapters introduce two of the resulting datasets that made it to production – the **FEVER CS** and the **ČTK** dataset – and the methods of their collection.

Property	**FEVER CS**	**ČTK**
Obtained through	Machine Translation	Annotation experiments
Language style	Encyclopædic	Journalistic
Retrieval unit	Sentence	Paragraph
Cross-references	First level links	Knowledge scopes (4.3.4)
Main focus	Document retrieval	NLI (for the time being)
Size	127,328 claims	3,295 claims

Table 2.1: Comparison of FEVER CS and ČTK datasets

[3] https://github.com/awslabs/fever

[4] https://github.com/HenriSchulte/Danish-Fact-Verification

Chapter 3
FEVER CS: a Dataset Localization

In Chapter 2, we have examined the existing datasets for our task. In this chapter, we will attempt to extract a part of the correct fact verification examples they carry, and localize them into Czech. More specifically, we will be proposing localization methods for the greatest one – the FEVER dataset.

Even though the localization process is prone to imperfections of all sorts, its resulting dataset will be of great use training the *baseline* models, as well as *pre-training* the finer models in the later stages of our work, when a native Czech dataset will be introduced for the *fine-tuning*.

3.1 FEVER format

Before we start to extract the Czech $(claim, evidence)$ pairs, let us examine the format of the FEVER datapoints.

```
{
  "id": 36242,
  "verifiable": "VERIFIABLE",
  "label": "REFUTES",
  "claim": "Mud was made before Matthew McConaughey was born.",
  "evidence": [
    [
      [52443, 62408, "Mud_-LRB-2012_film-RRB-", 1],
      [52443, 62408, "Matthew_McConaughey", 0]
    ],
    [
      [52443, 62409, "Mud_-LRB-2012_film-RRB-", 0]
    ]
  ]
}
```

Figure 3.1: Example FEVER `REFUTES` annotation with two possible evidence sets

- Dataset is stored in `JSON` Lines (`JSONL`) format, each line features a data point like 3.1 without the whitespace

- The verifiability is stored in attribute `verifiable` $\in \{\texttt{VERIFIABLE}, \texttt{NOT VERIFIABLE}\}$, veracity is stored using `label` $\in \{\texttt{SUPPORTS}, \texttt{REFUTES}, \texttt{NOT ENOUGH INFO}\}$

- `evidence` is a set of all possible *evidence sets*, any of these sets *alone* suffices to refute the `claim`
- Every such an *evidence set* is structured as a conjunction of Wikipedia sentences in format: `[annotation_id, evidence_id, article_wikiid, sentence_index]`

To illustrate the correct interpretation of data from the example 3.1, there are two possible counterproofs for the claim "*Mud was made before Matthew McConaughey was born.*":

Evidence set #1:

> [**Mud (film)**] Mud is a 2012 American coming-of-age drama film written and directed by Jeff Nichols.
>
> [**Matthew McConaughey**] Matthew David McConaughey (born November 4, 1969) is an American actor.

Evidence set #2:

> [**Mud (film)**] The film stars Matthew McConaughey, Tye Sheridan, Jacob Lofland, Sam Shepard, and Reese Witherspoon.

Figure 3.2: Evidence from data point 3.1

3.2 Localizing the FEVER

Before we introduce the single steps, let us design a simple scheme for localizing it into an arbitrary language exploiting the ties between FEVER and Wikipedia:

Starting from the FEVER [Thorne et al., 2018a] dataset:

1. Merge the FEVER train and dev datasets into a joint dataset fever_en
2. Using MediaWiki API, map every Wikipedia article used in fever_en evidences to its target localization, if none found, remove every *evidence set* that contains it to create the fever_lang – in our case, the fever_cs (Section 3.2.2)
3. Remove all `SUPPORTS` and `REFUTES` data points with empty evidence from fever_lang
4. Download the current Wikipedia dump in the target language, parse it into a *knowledge base* – plain text corpus keyed by article name (Section 3.2.1)
5. Localize every claim using the Machine Translation (Section 3.2.3)
6. Normalize every string value in fever_lang, as well as the knowledge base, using the same Unicode normal form (Section 3.2.4)
7. Sample around 0.05 · |fever_lang|[1] annotations for each label using a fixed *random seed*[2], store them as dev. Repeat for test.
8. Store the rest of labels as train

[1] This split size is proportional to that of FEVER EN – it balances the labels to punish bias and favours the train size, due to the data-heavy nature of the task

[2] This ensures the reproducibility

This scheme has notable weaknesses. Firstly, the evidence sets are not guaranteed to be exhaustive – no human annotations in the target language were made to detect whether there are new ways of verifying the claims using the target version of Wikipedia. Furthermore, an unknown number of evidence has lost its validity, as the given Wikipedia localization lacks the specific information needed to conclude the fact-checking verdict.

With all of the dataset's flaws listed above to keep in mind, it is an exciting starting point, appropriate for training and validating both the early Document Retrieval and Natural Language Inference models. Therefore, we argue, it is a fruitful experimental dataset for both fields of our research.

Now, let us reinforce on the non-trivial points from the scheme above, specifically for our Czech instance.

3.2.1 Czech Wikipedia (June 2020) corpus

As an experimental *knowledge base*, we are providing the CS June 2020 Wikipedia dump parsed into a plain text corpus using the WikiExtractor and structured into a FEVER-like knowledge base, i.e., a SQLite single-table[3] database, providing every article *abstract* from the CSWiki dump, structured as its `id`, `text` and its `sentences`-split, computed using the Punkt [Kiss and Strunk, 2006] sentence tokenizer.

The resulting knowledge base can be downloaded from our webpage[4] and the tools for its extraction are open-sourced in the section 5.7.

3.2.2 Localization data loss

For every article, Wikipedia provides a set of links to the same article in different foreign languages. This feature is powered by the MediaWiki database and can be accessed programatically through the MediaWiki API [Astrakhan et al., 2021].

In the early stage of development, we have written the ad–hoc localize_dataset[5] Python module to exploit this feature. Its outputs for the `cs` target language (measuring the *data loss* in the step 2. of 3.2) are *highly* encouraging:

```
Of 12633 articles: 6578 preserved, 6055 lost, of which 84 due to
↪ normalization
Of 145449 data points, 112969 survived the localization (of which
↪ 35639 was not verifiable), 32480 didn't
```

That means the majority of FEVER-adjacent articles *do* have their Czech translation, and, even more surprisingly, whole **78%** of claims can be fully (dis-)proven in at least one way using only the Czech Wiki. That is, in an ideal world where the Czech abstracts are semantically equivalent to their English counterparts and no `NOT ENOUGH INFO` annotations are lost due to a piece of knowledge unique to CSWiki.

Still, this is most often the case for the points we examined, and even though the original sentence indices from 3.1 can not be trusted, the `wikiid` typically can. The *precision/recall* (6.1.1) metrics are yet to be done using human annotations, however, the empirical intuition would be that *recall* took most of the damage (*evidence sets* "forgotten" by our dataset).

[3] The table name is *Document*

[4] `http://bertik.net/cswiki`

[5] `https://github.com/heruberuto/fact-checking/blob/master/localize_dataset.py`

3.2.3 Tested Approaches for English-Czech Machine Translation

As the Machine Translation model evaluation is a complex field of its own, and as the standard metrics such as BLEU [Papineni et al., 2002] or SacreBLEU [Post, 2018] require a number of human-annotated reference translations, we do not possess the time to properly cover it in our research project. Thus, we are down to our own empirical observations of the translation quality. Our conclusions should be taken as such.

From the tools openly available online in a ready-to-use manner, we have examined the following (Table 3.1):

1. **Google Cloud Translation API** [Google, 2021] was the platform we used to translate the first version of FEVER CS dataset, as it is convenient to use on a large set of data, and as it empirically yielded a *comprehensible* translation for the majority of claims.

2. **LINDAT Translation Service** [Košarko et al., 2019] uses CUBBITT [Popel et al., 2020] transformer model for machine translation and was released after its publication in Nature in September 2020. It performs on par with the top commercial-grade translators, however, it is published under a restrictive license for personal use.

3. **DeepL** [DeepL, 2021] released its English–Czech translation model for public use on March 17th 2021. While we found out about it two weeks before the thesis submission deadline, we feature its outputs in the final dataset, as we have observed its translations to be superior both in the *translation adequacy*[6] and the fluency of the resulting texts. We have found it to be very robust against homonyms[7], which is crucial for preserving the claim meaning and, therefore, the validity of transferred evidence.

Original claim	EN		**“Harald V of Norway married a commoner.”**
Google Translate	CS	April’20	“Norská Harald V se oženila s občanem.”
Google Translate	CS	May’21	“Harald V Norska si vzal prostého občana.”
CUBBITT	CS	May’21	“Harald V. z Norska si vzal neurozenou ženu.”
DeepL	CS	May’21	“Harald V. Norský se oženil s obyčejnou ženou.”
Original claim	EN		**“Indiana Jones has been portrayed by an actor.”**
Google Translate	CS	April’20	“Indiana Jones byl vylíčen hercem.”
Google Translate	CS	May’21	“Indiana Jones byl zobrazen hercem.”
CUBBITT	CS	May’21	“Indiana Jonese ztvárnil herec.”
DeepL	CS	May’21	“Indiana Jones byl ztvárněn hercem.”
Original claim	EN		**“Manchester by the Sea has grossed money.”**
Google Translate	CS	April’20	“Manchester u moře rozdal peníze.”
Google Translate	CS	May’21	“Manchester by the Sea vydělal peníze.”
CUBBITT	CS	May’21	“Manchester by the Sea utržil peníze. ”
DeepL	CS	May’21	“Film Manchester by the Sea vydělal peníze.”

Table 3.1: Machine Translator comparison using FEVER claims. Examples were cherry-picked to highlight the observed differences between translators.

[6] Preserving text meaning [Popel et al., 2020]

[7] Words that can have different meanings (and therefore different Czech translations), which, typically, must be guessed from the context – s. a. the “river *bank*” and the “retail *bank*”

3.2.4 Unicode Normalization

In the Unicode paradigm, the same diacritized character can be stored using several different representations [Whistler, 2020]. To allow straightforward byte-wise comparisons on the low level (s. a. TF-IDF search, neural networks, …), one should eliminate this property using one of the *Unicode normal forms*:

1. **NFD** – *Normalization Form Canonical Decomposition* – fully expands out the character (see Figure 3.3). Is faster to compute, but ultimately takes more space. Theoretically, it could be used to exploit the property of Czech, that the words that have similar undiacritized representations tend to be semantically close (e. g. "být́i" and "bytí" share 4 bytes rather than 2 in NFD)
2. **NFC** – *Normalization Form Canonical Composition* – runs the NFD algorithm and then combines the characters where possible – it runs longer, but the resulting strings take up less space

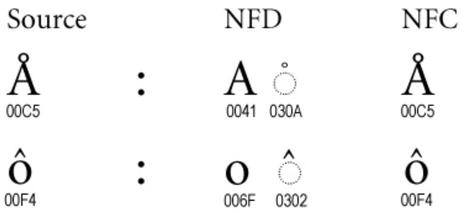


Figure 3.3: Unicode normal forms, reprinted from [Whistler, 2020]

3.2.5 Document leakage

An interesting and possibly malicious property of our dataset is a large *document leakage* due to the simplistic splitting strategy defined in 3.2 step 9.

Simply put, the train and test splits may contain claims related to the same evidence-set document. However, this was neither addressed by [Thorne et al., 2018a], as we have found **11,165** out of their **13,332** dev[8] annotations sharing an evidence-set document with some train claim.

A further research is needed to answer whether this is a problem and what are the optimal strategies to punish model *overfitting* while still optimizing for a broad topic coverage, proportional to the number of leakage-prone documents.

3.3 Resulting Dataset

	FEVER CS			**FEVER EN**		
	SUPPORTS	REFUTES	NEI	SUPPORTS	REFUTES	NEI
train	53,542	18,149	35,639	80,035	29,775	35,639
dev	3,333	3,333	3,333	6,666	6,666	6,666
test	3,333	3,333	3,333	6,666	6,666	6,666

Table 3.2: Label distribution in FEVER CS dataset as oposed to the FEVER EN

[8] Only the SUPPORTS and REFUTES annotations are considered in our measure, as the NEIs do not carry any evidence to compare against.

In Table 3.2 we show the label distribution in our dataset, is roughly proportional to that in FEVER EN. Inspired by the [Thorne et al., 2018a] paper that only uses a dev, test of 3,333 claims per annotation to establish the baseline models, we have opted the same split size. This decision was experimental and should be further challenged in the future.

Following the scheme described in 3.2, we have released its open source implementations[9] [10] for an arbitrary language, and a set of ready-made train, test and dev data[11] in its most recent version Machine-Translated by DeepL. Both the data and the implementations are being published under the CC BY-SA 3.0 license.

[9] `https://github.com/aic-factcheck/fever-cs-dataset`

[10] `https://github.com/heruberuto/fact-checking/`

[11] `http://bertik.net/fever-cs`

Chapter 4
ČTK Dataset Collection

In Chapter 3, we have acquired the initial fact-checking dataset in Czech via localizing that of [Thorne et al., 2018a]. The localized dataset relies on the Wikipedia dump as its knowledge base.

As the Wikipedia does not call itself a reliable source for a multitude of reasons [Wikipedia, 2021b], a further research is desirable on how to transfer the fact verification practice learned on FEVER to a whole other knowledge base.

This raises a variety of interesting challenges, in particular: how to transition away from the encyclopædic style [Wikipedia, 2021a] of written language? Could one transfer the fact-checking rules learned on such a strictly formatted corpus to, say, an archive of news reports, with all of its pitfalls, such as the *temporal reasoning*[1]?

4.1 Academic Cooperations

As a part of the project *"Transformation of Ethical Aspects With the Advent of Artificial Intelligence Journalism"* funded by the Technology Agency of the Czech Republic (TAČR), we have been given an opportunity to work with Václav Moravec from the Faculty of Social Sciences, Charles University (FSS), and the students of his courses in *AI Journalism*.

Furthermore, we have been granted access to the full archive of Czech News Agency (ČTK), that, by the time of creating a snapshot, contained a total of 15,032,152 news reports released between 1st January 2000 and 6th March 2019[2], which we have reduced to the size of 11,134,727 reports by removing the sport results and daily news summaries.

Thanks to these cooperations, we have been offered to work with around 170 human annotators, mostly undergraduate and graduate students of FSS. During three "waves" of annotation, we have collected a total of 10,084 data points (3,293 original claims and their respective labels and evidence).

In this chapter, we would like to describe how we tailored our own annotation platform to the needs of our task and annotators, justify the design choices that we made, and summarize our experience of supervising three waves of human claim generation and labeling experiment.

[1] Typical case would be a journal archive containing two mutually exclusive *ground truths* different in their timestamps, s. a. "Summer 2018 was the warmest" and "Summer 2019 was the warmest"

[2] Efforts are being made to re-insantiate the following data collection experiments on an extended version of the archive, up to December 2020, so as to cover the topic of COVID-19 pandemic. These were, however, postponed subsequent to this thesis, in order to maintain its data consistency.

4.2 Requirements for the Annotation Platform

Before we unravel the solutions provided by our platform, let us first spend a section to establish the underlying problems. Even though we do not follow any strict procedure of the *requirements modelling* [Nuseibeh and Easterbrook, 2000], we believe that what follows are the most important challenges our system should tackle and other researchers building such a tool might want to keep in mind:

1. **FEVER-like annotation tasks** – despite the corpus differences, we aim to follow the concept-proven task division from [Thorne et al., 2018a]:

WF1a **Claim Extraction** provides annotator A with a document d from the Wiki corpus, A outputs a simple factoid claim c extracted from d without using A's own world knowledge

WF1b **Claim Mutation**: feeds c back to A, who outputs a set of mutations of c: $M^c = \{m_1^c, \dots m_n^c\}$ using A's own world knowledge (*negation*, *generalization*, ...). For the completeness, we reprint the Table 4.1 that lists the allowed types of mutations.

Type	Claim	Rationale
Rephrase	President Obama visited some places in the United Kingdom.	Rephrased. Same meaning.
Negate	Obama has never been to the UK before.	Obama could not have toured the UK if he has never been there.
Substitute Similar	Barack Obama visited France.	Both the UK and France are countries
Substitute Dissimilar	Barrack Obama attended the Whitehouse Correspondents Dinner.	In the claim, Barack Obama is visiting a country, whereas the dinner is a political event.
More specific	Barrack Obama made state visit to London.	London is in the UK. If Obama visited London, he must have visited the UK.
More general	Barrack Obama visited a country in the EU.	The UK is in the EU. If Obama visited the UK, he visited an EU country.

Table 4.1: FEVER Annotation Platform mutation types – the examples mutate the claim "Barack Obama toured the UK" – reprinted from [Thorne et al., 2018a]

WF2 **Claim Labeling**: A is given a sample of a mutated claim m^c, context that was given to extract c in (a.) and is tasked to output sets of evidence $E_1^{m^c}, \dots, E_n^{m^c}$ along with the veracity label $g(E_i^{m^c}, m^c)$ which should be the same for each i.

Apart from the context of c, A can fulltext search the entire Wikipedia for evidence, however, A operates under constrained time.

2. **Paragraph-level documents** – the FEVER shared task proposed a *two-level* retrieval model: first, a set of *documents*, i.e., Wiki abstracts is retrieved, then these are fed to the *sentence retrieval* system which retrieves the evidence on the level of sentences.

   This simply does not work for us – firstly, the sentences of a news report corpus are significantly less *self-contained* than those of encyclopædia abstract, not supporting the *sentence*-level of granularity. Secondly, the articles tend to be overly long for the Document Retrieval task.

We argue that the best approach for our data is the *paragraph-wise* splitting and a single level of retrieval, with an option of grouping a set of paragraphs by their source article. From this point we refer to the ČTK paragraphs also as to the *documents*.

3. **Source document sampling system** – in [Thorne et al., 2018a], every claim was extracted from some sentence sampled from a Wikipedia abstract. With news report archive, this does not work well, as the majority of ČTK paragraphs does not contain an information eligible for fact-checking.

4. **Limited knowledge oracle access** – in FEVER **Claim Extraction** as well as in the annotation experiment of [Binau and Schulte, 2020], the annotator was provided with a Wikipedia abstract and a *dictionary* composed of the abstracts of articles *linked* in it. This was important to ensure that the annotators only incorporate their full world knowledge in a restricted number of well defined tasks, and limit themselves to the facts (dis-)provable using the corpus in the rest.

   As the ČTK corpus does not follow any rules for internal linking, this will be a major challenge to reproduce.

5. **Annotator performance measures** – completion of the annotation tasks is going to count towards the completion of the FSS course. Therefore, the annotator's identity needs to be stored within the system, and a set of reasonable goals must be proposed to track the completion of student's duties.

   Reaching the goals should take under 3 hours on average, which matches the share of the annotation assignment on the ECTS study load of the FSS course [The European Commission, 2015].

6. **Cross-annotator validation** – to measure the validity of the annotations, as well as that of our novel platform, a claim should be labeled more than once on average.

   This will allow us to quantify the inter-annotator agreement, as well as it increases the overall number of evidence sets per claim. We do not consider the task of enumerating *every* possible evidence set from the ČTK corpus feasible, as the news archives are not limited in the number of duplicate information they store. However, the more the better.

7. **ČTK Archive access mediation** – Due to the size of the ČTK archive (~11M reports with metadata) and our space constraints that do not allow a very generous indexing, we need a caching system that only stores the articles necessary for annotating the claims that are currently in the system. This reduces the lookup time and increases the maximum traffic load.

4.3 FCheck Platform

In Figure 4.1, we model the basic structures of data our system is working with and their relations using the standard entity–relationship diagram [Chen, 1976].

In contrast with the simplicity of the structured JSONL annotation format shown in the Figure 3.1, our data model is rather complex. Our aim here is to exploit the properties of relational database to find annotation ambiguities and easily compute the annotator performance measures through SQL *aggregations*

In addition, we introduce the automatically updated UNIX timestamps `created_at` and `updated_at` to every entity from 4.1, to be able to reconstruct the annotation timeline, as well as to generate a dashboard of live visualisations of the performance and validity metrics, examples of which are the Figure 4.7 and 4.6.

4.3.1 Entities and Their Relations

Every annotator is to be identified through their respective *User* object, storing the necessary credentials and signing every annotated data-point with their identifier. The data-points are divided into *Claim*s and *Label*s. Each *Claim* is either extracted from a ČTK *Paragraph*, linked as `paragraph`, or from a parent *Claim*, linked as `mutated_from`.

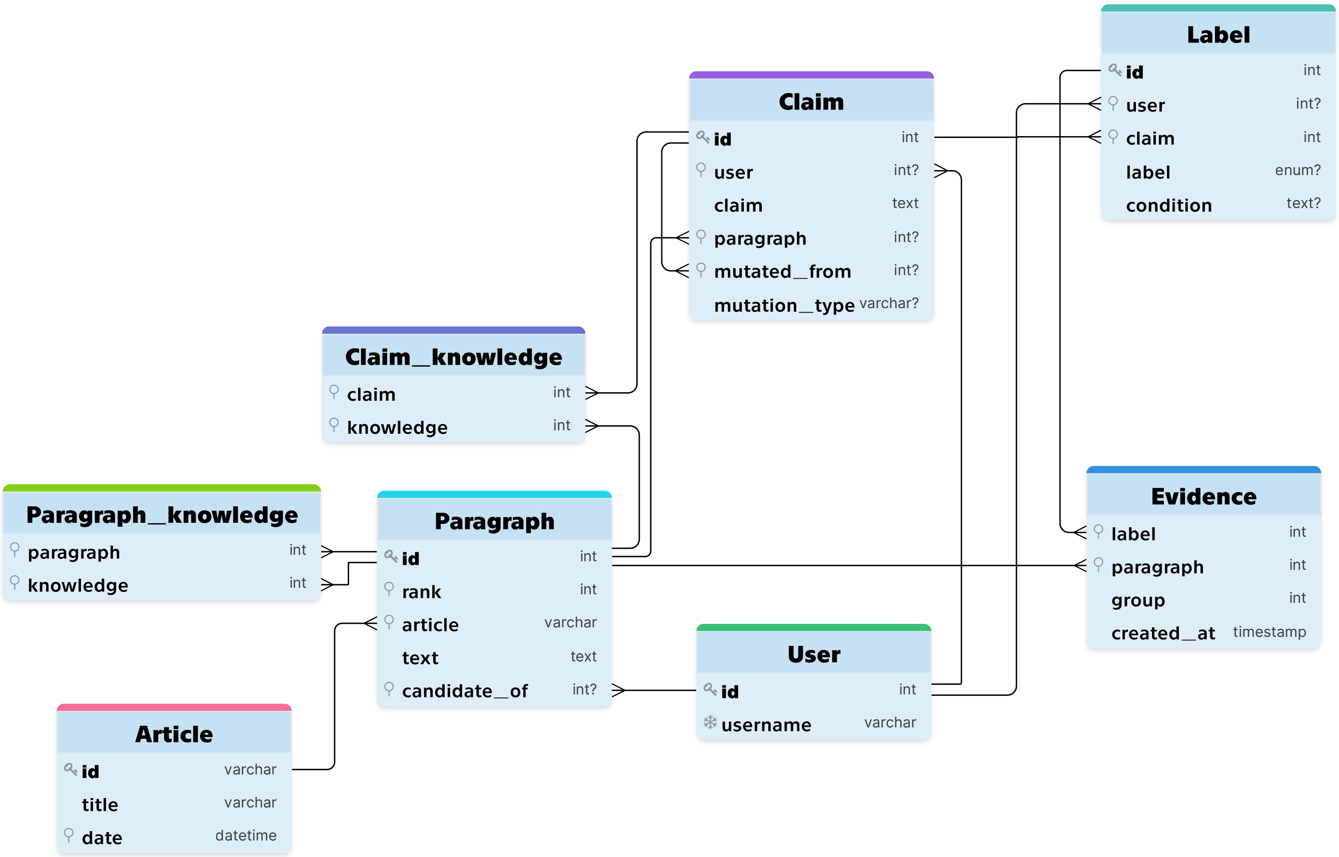


Figure 4.1: Entity–relationship diagram of the FCheck application drawn using [`drawSQL`, 2021]

For the Claim Labeling and Claim Extraction tasks, user is to be given a restricted scope of knowledge. This knowledge can be described as a set of *Paragraph* objects, and is defined for a given *Claim* or *Paragraph* (*many-to-many* relations *Paragraph_knowledge* and *Claim_knowledge*) – we will define the *knowledge scopes* in 4.3.4.

The *Label* data-point is characterized by the `label` itself (*enum* of `SUPPORTS`,`REFUTES`,`NOT ENOUGH INFO`) and its *Evidence* – a set of *evidence sets*. Each such evidence set consists of *Paragraph*s and is distinguished by a different ordinal number `group` stored alongside the *Evidence* relation. Therefore, a single *Label* can have multiple alternate sets of evidence, just as demonstrated in the Figure 3.1.

Several complementary entities were hidden away for the simplicity of the 4.1, however, are not integral to the data model of our application – for example the *annotation* *stopwatch* data.

4.3.2 Technology Stack

Originally, we have planned to re-use the Flask annotation platform of [Thorne et al., 2018a] with minor tweaks. Sadly, we were unable to fully recover the open source version[3], as there was static data of unknown structure missing for Wiki *redirects*.

Even so, these efforts would have been rendered futile by the invention of *knowledge scopes* that we will introduce in 4.3.4.

Thus, we have embarked on the journey to build our very own annotation platform, heavily inspired by that of [Thorne et al., 2018a], using our preferred technologies:

1. **PHP 7** will be running the annotation back-end, written using the **Yii2** framework, served by **Apache2** on **Debian**
2. **MySQL 8** is to be storing the entities from 4.1 in form of the SQL tables
3. **Python 3.7**, **PyTorch**, **Flask** and **SQLite3** provide an API for a direct ČTK data access, as well as to the neural networks and clustering required for semantic search
4. **AJAX** will be used to asynchronize the API calls, so that a user can keep annotating on the Apache server while the computation-heavy tasks are being processed by Flask

Despite the choice of technologies does not follow the most recent trends, we have decided for it because of its familiarity. As the annotation leadership and administration are tasks heavy on technical support and hotfixing, we favoured the tools we have several years of commercial experience with.

4.3.3 Corpus Caching: Proxy Entities

The *Article* and *Paragraph* db entities from 4.1 serve as a proxy for the slowly attainable entries of the full ČTK corpus which is stored separately and its paragraphs are copied to the FCheck database on demand.

The idea is that if we provide a background service that asynchronously precomputes which paragraphs of the full corpus are to be provided to an annotator for the given task and input data, we can simply copy them into a well-indexed smaller database integrated with the rest of the system through a relational database.

Thus, we were able to scale down the amount of data hardwired to the interactive part of the platform from ~10^8 to the order of 10^4 paragraphs, dramatically improving the lookup times while also obtaining a compact *self-contained* database that can be easily backed up and still contain all the corpus entries necessary for exporting the dataset.

4.3.4 Knowledge Scopes

In place of Wikipedia dictionaries that were used used in FEVER annotation task, we propose the following framework for knowledge delimitation:

We have used the DrQA [Chen et al., 2017] and multilingual BERT [Devlin et al., 2019] models trained by [Pitr, 2020] during his summer AIC internship as our internal state-of-the-art for FEVER CS wiki-abstract retrieval. The model task was to output a set of k semantically nearest paragraphs (k-NN) to the given string.

[3] `https://github.com/awslabs/fever/tree/master/fever-annotations-platform`

Where DrQA operates on a verbatim (*term frequency–inverse document frequency*) basis, mBERT model calculates the paragraph *embeddings* using a Transformer network pretrained on Wikipedia and finetuned for the Czech Document Retrieval task.

To have the best of both worlds, we have used a simple ČTK Archive Flask API, implemented by [Drchal and Ullrich, 2020] as a *façade* that receives a claim (or a paragraph identifier) through a HTTP request, and responds with a combination of the results of both of the aforementioned models.

ČTK Archive Flask API

Is a simple HTTP API we have co-authored with our supervisor Jan Drchal. It encapsulates the access to the ČTK Archive corpus via *random sampling* and the *Knowledge scope* enumeration, which follows:

In brief words, it makes multiple calls to the DrQA, fortifying the claim by a different pair of mentioned *named entities* in each[4], to obtain their highest-utility results for each NE pair. Then it picks the 4 documents with the *overall* highest utility as the `search_ner` result. The Named Entity Recognition is handled by the model of [Straková et al., 2019].

It then retrieves the 1024 top documents for the query using mBERT, and clusters their embeddings into 2 groups with k-means. Then, two closest representatives of the *claim embedding* from every cluster are stored as the `search_semantic`.

The flask ouputs both `search_ner` and `search_semantic`, i.e., a maximum of 8 documents per query, not to overwhelm the annotator. Furthermore, it makes sure that all the retrieved paragraphs have an older timestamp than the input. This outlines our solution for the *temporal reasoning* issue. Simply put, to each claim, we assign a date of its formulation, and only verify it using the news reports published *to that date.*

Using the example from the chapter introduction, the paragraph "Summer 2019 was the warmest" (say, published at Septamber 24th, 2019) will only be considered a *ground truth* for claims with a timestamp $\geq$ `2019-09-24`. For the completeness, later, in the task $\mathsf{T}_{\mathtt{1b}}$, we assign each claim with the publication timestamp of its source paragraph.

4.4 The Annotation Workflow

In the previous sections, we have explained the technical challenges and their respective solutions. An equally important task is that of supervising a group of annotators new to this system and streamlining a sequence of tasks that both guides the annotators to the best use of their expertise in journalism and saturates the dataset.

[4] E. g. the claim "Miloš Zeman visited Slovakia." is augmented by an extra copy of the entity "Miloš Zeman", and "Slovakia", to boost their *term frequency.*

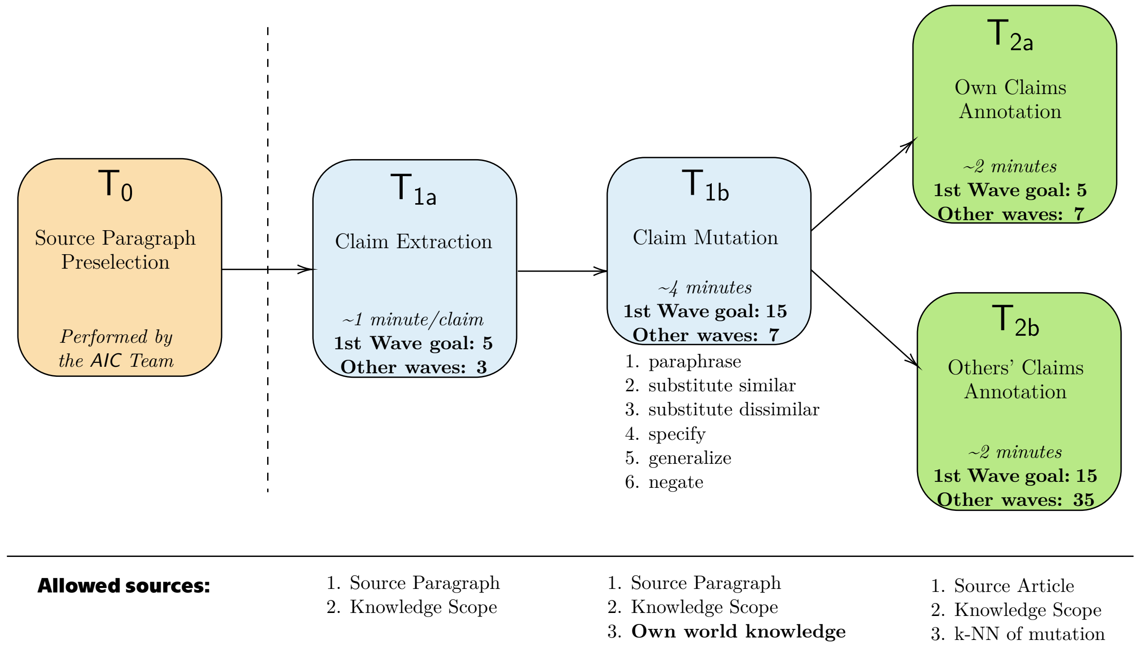


Figure 4.2: Annotation workflow diagram presented to the FSS annotators, redrawn from [Drchal, 2020]

4.4.1 Revised Annotation Tasks

To satisfy our requirements, we have adjusted the annotation tasks from Section 4.2 in the following ways:

For reader's convenience, we mostly use a simplified set of actors – Flask is the "slow" back-end API, that operates above with the full ČTK Archive and models from 4.3.4, Apache is a lighter web interface above the entities of 4.1 accessible to A, A is the annotator.

T_0 **Source Paragraph Preselection:** Flask samples a source article, Apache caches it (see 4.3.3). A spends ≤ 30 seconds skimming the article and, finally, nominates a single paragraph p to be used in T_{1a}.

p must feature a *self-contained* piece of verifiable information. If there is no such paragraph, A skips to the next sample.

Otherwise, Apache stores the nomination and Flask enqueues the *knowledge scope* computation for p. Once finished, result will be forwarded to Apache, which will cache the retrieved paragraphs and their respective articles and store them as $knowledge(p)$.

T_{1a} **Claim Extraction:** Apache samples a nominated paragraph p, provides A with p and $knowledge(p)$. A outputs a simple factoid claim c extracted from $\{p\} \cup knowledge(p)$ without using A's own world knowledge

T_{1b} **Claim Mutation**: Apache feeds c back to A, who outputs a set of mutations of c: $M^c = \{m^c_1, \dots m^c_n\}$ using A's own world knowledge (*negation*, *generalization*, ...)

To catch up with the additional knowledge introduced by A, Flask enqueues the computation of $\{knowledge(m^c_1), \dots knowledge(m^c_n)\}$, and, once done, notifies Apache to store these, as well as to cache the incident paragraphs.

T_{2a} **Own (Oracle) Claim Labeling**: Apache samples a fresh m^c made by A, and provides its source paragraph p, its full original article, and a shuffled set of articles from $knowledge(m^c) \cup knowledge(p)$.

A spends ≤ 3 minutes looking for the evidence sets $E_1^{m^c}, \dots, E_n^{m^c}$ along with the veracity label $g(E_i^{m^c}, m^c)$. Apache saves them as an *oracle annotation*.

T_{2b} **Others' Claim Labeling**: same as T_{2a} for m^c made by *other* annotator than A. Stored as *regular annotation*.

4.4.2 Conditional annotations

During our tests of the interface, a common problem with the annotation task T_2 using the ČTK corpus was that of *assuming the knowledge*. Using the mutation types such as *generalization*, one would often run into generating a claim containing a mutation not fully provable using a news archive.

For example, for a claim "Miloš Zeman did not visit an European country", system 4.3.4 often retrieves relevant knowledge s. a. "Miloš Zeman visited Slovakia". However, it barely ever retrieves the neccessary conclusive proof that "Slovakia is a European country".

To address this issue, we are introducing the concept of *conditional annotations*: if the annotator can not construct an exhaustive evidence set, but possesses knowledge that would *conclude* the *partial* set of evidence, he is asked to write it down in a form of *textual claim* $c_{condition}$. Then, if any annotator could `SUPPORT` the $c_{condition}$ using a freshly computed $knowledge(c_{condition})$, it would also yield the paragraphs that would *complete* the partial sets of the original evidence.

Claim: "The Killers performed in **the second day** of Rock for People 2007."
Label: `REFUTES`
Condition: "The first and the second day of Rock for People had disjoint line-ups."

Evidence set #1:

The first day of Rock for People culminated with the concert of The Killers [July 4^{th} 2007]

Hradec Králové, 4th of July (ČTK) - Today, an hour before midnight at the Hradec Králové airport, American guitar band The Killers performed their concert, which was the climax of **the first day** of the festival. Above the mucisians' heads hanged a shining sign "Sam's Town", which is the name of their second album that came out last year. The musicians came to introduce the songs from this album to the festival audience.

Figure 4.3: Example of a conditional label (translated from the ČTK v2.1 dataset). See that if we are able to `SUPPORT` the **condition**, we can use the union of any of its evidence-sets together with the set *#1* to disprove the original claim. If not, the correct label is `NEI`.

4.5 Web User Interface

Finally, we are including a look into the client-side of the annotation platform we have presented to our annotators.

In 4.2, we have estimated a maximum time of **3 hours** to complete the entire annotation workflow (4.5). Of these, we have dedicated the first **30 minutes** to a **video-tutorial**[5],

[5] `https://fcheck.fel.cvut.cz/site/tutorial` or `https://youtu.be/AcarF4Rxexc`

which, despite its length, worked well[6] in giving every annotator a full platform walk-through and a hands-on example for every task.

Therefore, each student should spend only **2.5 hours** on our platform to reach all the annotation goals from Figure 4.5. This puts pressure on the design of the client-side, to be as easy to use as possible, while still supporting the sophisticated features, s. a. the *conditional-* and *multi-annotation*. In this chapter, we will show the web interface for the main tasks, and justify the design choices made.

To our readers, we also provide the link to the live[7] platform which can be found at `https://fcheck.fel.cvut.cz` and accessed by typing `testuser` into the "SIDOS ID" field. Do not worry to play around, as all of the `testuser`'s annotations can be easily omitted from the exports.

4.5.1 Claim Extraction

We present our final T_{1a} interface in Figure 4.5. The layout is inspired by the work of [Thorne et al., 2018a] and, by default, hides as much of the clutter away from the user as possible. Except for the article heading, timestamp and the source paragraph, all the information such as the *knowledge base* is collapsed and only rendered on user's demand.

During the first run, user is instructed to read through detailed **Instructions** in a Bootstrap *modal* window, that, for the rest of the time, stay hidden away again not to distract the annotator. Apart from these, we have only added a brief instruction to each the form field as a reminder.

Annotator reads the source article and, if it lacks a piece of information he wants to extract, looks for it in the expanded article or knowledge base entry. Extracted claims are to be typed into a HTML textarea and separated by the line break, which was the most intuitive method we experimented with. User is encouraged to **Skip** any source paragraph that is hard to extract.

Throughout the platform, we have ultimately decided not to display any *stopwatch*-like interface not to stress out the user. However, there is a simple tracking JavaScript running in the background, storing time spent on each page. From this data, we have measured that, excluding the outliers ($\leq 10s$, typically the **Skip**ped annotations and $\geq 600s$, typically a browser tab left unattended), average time spent on this task is **2 minutes 16 seconds** and the median is **1 minute 16 seconds**.

After the first wave of annotation, we have augmented the interface with a triple of *golden rules* to avoid repeating the most common mistakes. More on that in 4.6.1.

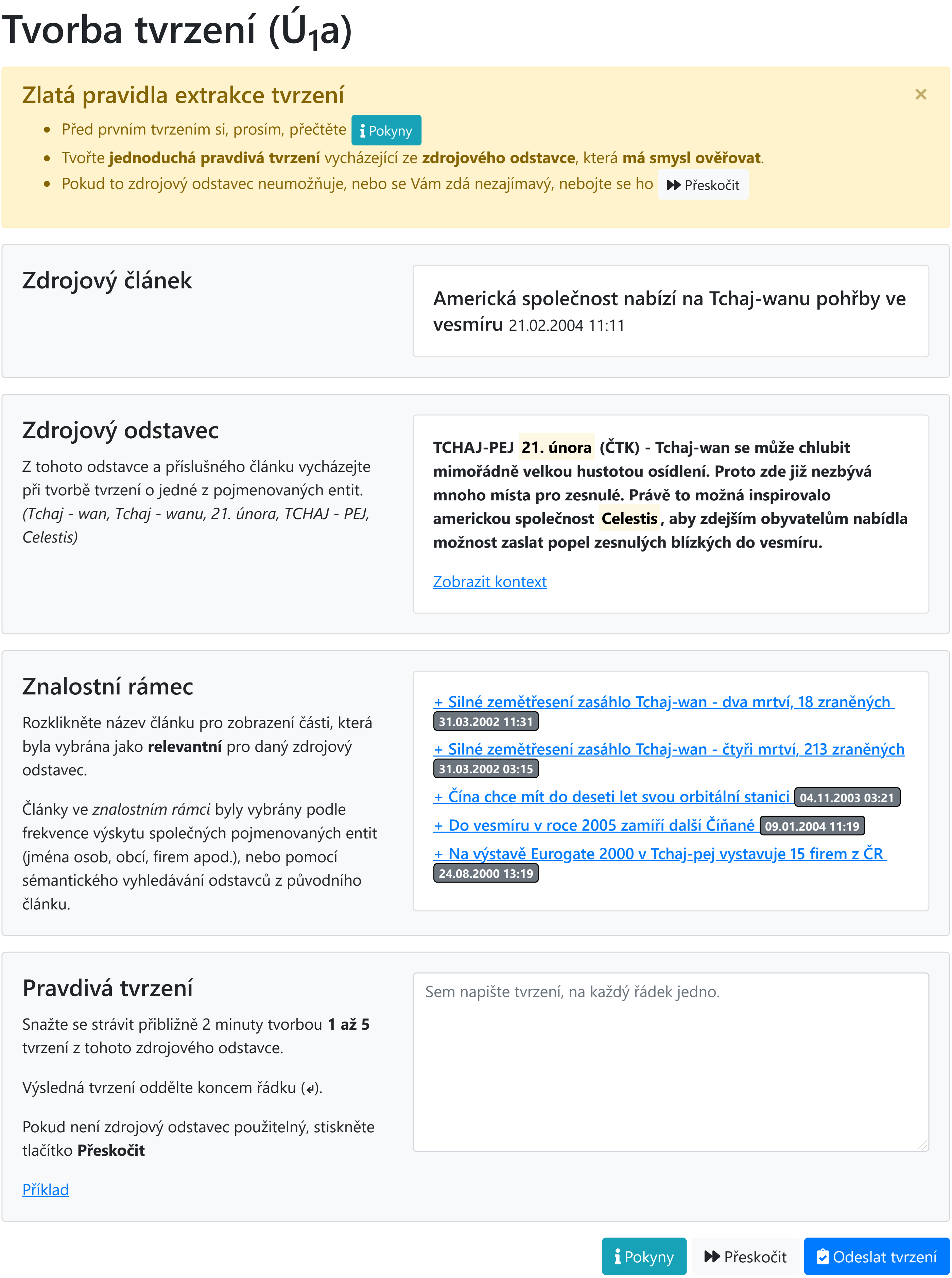
4.5.2 Claim Mutation

Follows the UI conventions set by the Claim Extraction. Mutation types follow those of the FEVER Annotation Platform (Table 4.1) and are distinguished by loud colors, to avoid mismatches. The mutation types will be a topic for further innovations in future, as our annotation experiments did not yield a label-balanced dataset.

Excluding the outliers, the overall average time spent generating a batch of mutations was **3m 35s** (median **3m 15s**) with an average of **3.98** mutations generated per claim.

[6] After refining the tutorial and the supplementary lecture after the first wave of annotations, we have observed a significant decrease in the task procrastination (see Figure 4.6), which may also have been caused by other factors. However, it had a good impact on the traffic spread, as well as on the quality of T_2 sampling (Figure 4.7).

[7] For as long as the FEE CTU keeps providing us with the computing power…



Tvorba tvrzení (Ú₁a)

Zlatá pravidla extrakce tvrzení

- Před prvním tvrzením si, prosím, přečtěte ℹ Pokyny
- Tvořte **jednoduchá pravdivá tvrzení** vycházející ze **zdrojového odstavce**, která **má smysl ověřovat**.
- Pokud to zdrojový odstavec neumožňuje, nebo se Vám zdá nezajímavý, nebojte se ho ⏩ Přeskočit

Zdrojový článek

Americká společnost nabízí na Tchaj-wanu pohřby ve vesmíru 21.02.2004 11:11

Zdrojový odstavec

Z tohoto odstavce a příslušného článku vycházejte při tvorbě tvrzení o jedné z pojmenovaných entit. *(Tchaj - wan, Tchaj - wanu, 21. února, TCHAJ - PEJ, Celestis)*

TCHAJ-PEJ 21. února (ČTK) - Tchaj-wan se může chlubit mimořádně velkou hustotou osídlení. Proto zde již nezbývá mnoho místa pro zesnulé. Právě to možná inspirovalo americkou společnost Celestis, aby zdejším obyvatelům nabídla možnost zaslat popel zesnulých blízkých do vesmíru.

Zobrazit kontext

Znalostní rámec

Rozklikněte název článku pro zobrazení části, která byla vybrána jako **relevantní** pro daný zdrojový odstavec.

Články ve *znalostním rámci* byly vybrány podle frekvence výskytu společných pojmenovaných entit (jména osob, obcí, firem apod.), nebo pomocí sémantického vyhledávání odstavců z původního článku.

- + Silné zemětřesení zasáhlo Tchaj-wan - dva mrtví, 18 zraněných 31.03.2002 11:31
- + Silné zemětřesení zasáhlo Tchaj-wan - čtyři mrtví, 213 zraněných 31.03.2002 03:15
- + Čína chce mít do deseti let svou orbitální stanici 04.11.2003 03:21
- + Do vesmíru v roce 2005 zamíří další Číňané 09.01.2004 11:19
- + Na výstavě Eurogate 2000 v Tchaj-pej vystavuje 15 firem z ČR 24.08.2000 13:19

Pravdivá tvrzení

Snažte se strávit přibližně 2 minuty tvorbou **1 až 5** tvrzení z tohoto zdrojového odstavce.

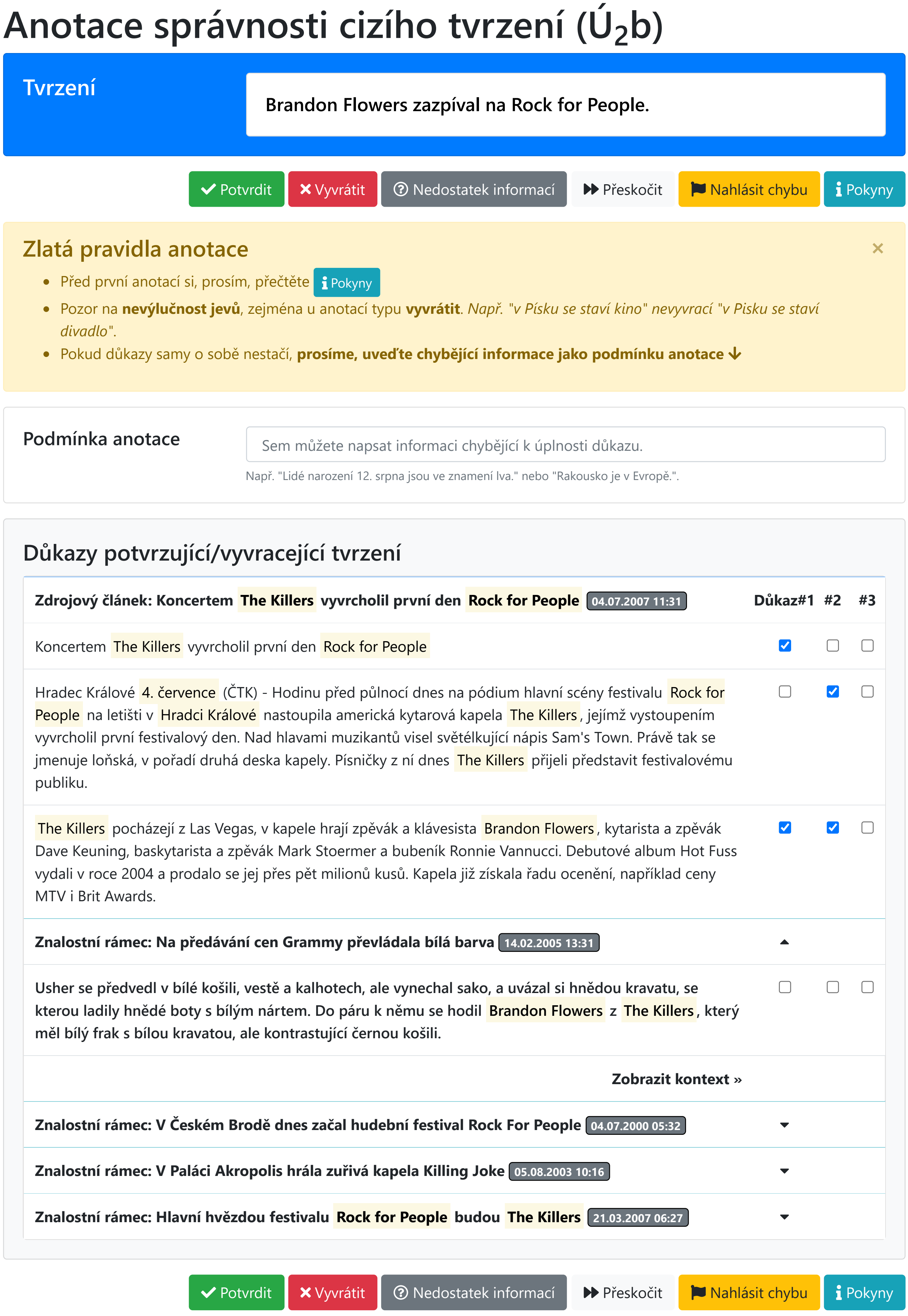
Výsledná tvrzení oddělte koncem řádku (↵).

Pokud není zdrojový odstavec použitelný, stiskněte tlačítko **Přeskočit**

Příklad

Sem napište tvrzení, na každý řádek jedno.

Figure 4.4: Claim extraction interface of the FCheck platform



Anotace správnosti cizího tvrzení (Ú$_2$b)

Tvrzení

Brandon Flowers zazpíval na Rock for People.

Zlatá pravidla anotace

- Před první anotací si, prosím, přečtěte Pokyny
- Pozor na **nevýlučnost jevů**, zejména u anotací typu **vyvrátit**. *Např. "v Písku se staví kino" nevyvrací "v Pisku se staví divadlo".*
- Pokud důkazy samy o sobě nestačí, **prosíme, uveďte chybějící informace jako podmínku anotace ↓**

Podmínka anotace

Sem můžete napsat informaci chybějící k úplnosti důkazu.

Např. "Lidé narození 12. srpna jsou ve znamení lva." nebo "Rakousko je v Evropě.".

Důkazy potvrzující/vyvracející tvrzení

Zdrojový článek: Koncertem The Killers vyvrcholil první den Rock for People 04.07.2007 11:31	**Důkaz#1**	**#2**	**#3**
Koncertem The Killers vyvrcholil první den Rock for People	☑	□	□
Hradec Králové 4. července (ČTK) - Hodinu před půlnocí dnes na pódium hlavní scény festivalu Rock for People na letišti v Hradci Králové nastoupila americká kytarová kapela The Killers, jejímž vystoupením vyvrcholil první festivalový den. Nad hlavami muzikantů visel světélkující nápis Sam's Town. Právě tak se jmenuje loňská, v pořadí druhá deska kapely. Písničky z ní dnes The Killers přijeli představit festivalovému publiku.	□	☑	□
The Killers pocházejí z Las Vegas, v kapele hrají zpěvák a klávesista Brandon Flowers, kytarista a zpěvák Dave Keuning, baskytarista a zpěvák Mark Stoermer a bubeník Ronnie Vannucci. Debutové album Hot Fuss vydali v roce 2004 a prodalo se jej přes pět milionů kusů. Kapela již získala řadu ocenění, například ceny MTV i Brit Awards.	☑	☑	□
Znalostní rámec: Na předávání cen Grammy převládala bílá barva 14.02.2005 13:31	▲		
Usher se předvedl v bílé košili, vestě a kalhotech, ale vynechal sako, a uvázal si hnědou kravatu, se kterou ladily hnědé boty s bílým nártem. Do páru k němu se hodil Brandon Flowers z The Killers, který měl bílý frak s bílou kravatou, ale kontrastující černou košili.	□	□	□
Zobrazit kontext »			
Znalostní rámec: V Českém Brodě dnes začal hudební festival Rock For People 04.07.2000 05:32	▼		
Znalostní rámec: V Paláci Akropolis hrála zuřivá kapela Killing Joke 05.08.2003 10:16	▼		
Znalostní rámec: Hlavní hvězdou festivalu Rock for People budou The Killers 21.03.2007 06:27	▼		

Figure 4.5: Claim labelling interface of FCheck platform. Full English translation attached as Figure A.2

4.5.3 Claim Veracity Labeling

In Figure 4.5 we show the most complex interface of our platform – the T_2: **Claim Annotation** form.

Full instructions take about 5 minutes to read and understand, and are hid away in the **Instructions** modal window, that is to be opened during the first annotation, and the on demand. All actions are spread out on the top bar and *label condition* is collected through a text field above the evidence input. This was decided after a negative experience with using a modal **Actions** to hide away less frequent actions, originally inspired by other annotation platforms (see 4.6.2).

The input of multiple evidence sets works as follows: each column of checkboxes in 4.5 stands for a single evidence set, every paragraph from the union of knowledge belongs to this set iff its checkbox in the corresponding column is checked. Offered articles & paragraphs are collapsible (without loss of checkbox state), empty evidence set is omitted. Through JavaScript, interface always displays all the non-empty sets defined so far, plus one empty column of checkboxes that can be used to initialize a new one.

On average, the labelling task took **65 seconds**, with a median of **40s**. An average SUPPORTS/REFUTES annotation was submitted along with **1.29** different evidence sets, 95% of which were composed of a single paragraph – full histograms will be introduced in Chapter 5.

4.6 Between-Wave Platform Adjustments

Annotation wave is our term for a group of FSS students annotating towards a common deadline, for a fixed period of 10–14 days.

To date, have supervised a total of a 4 annotation waves, however, as the wave 3 and 4 were largely simultaneous and their deadlines only differed in two days, we group them together as the 3rd wave (Figure 4.6).

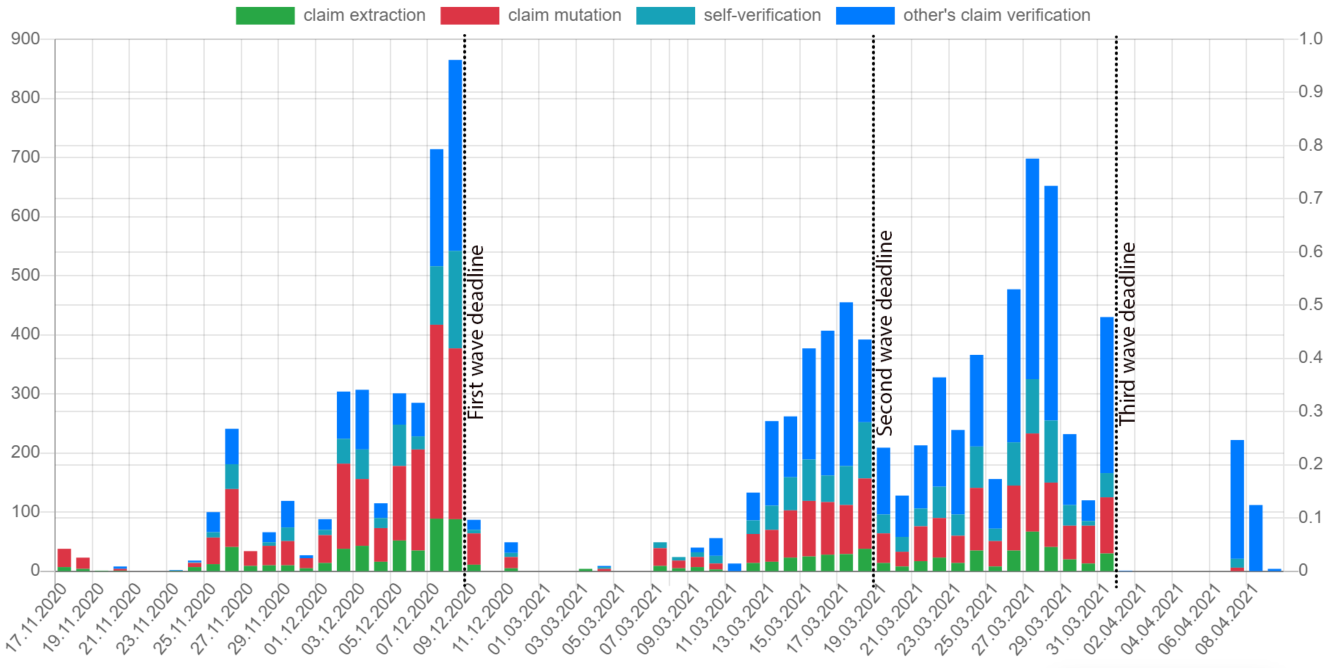


Figure 4.6: Number of data points generated per day, colored by task

Thanks to the multi-wave system of annotation, we were able to utilize our findings from the live data directly to patch the deployed version of the FCheck platform. This was particularly useful between the 1^{st} and the 2^{nd} wave. Here are the adjustments we presented, based on the learnings from the data exploration:

4.6.1 The Golden Rules of Each Task

To address the most frequently reoccurring annotation errors, which will be examined in depth in Chapter 5, we have came up with a set of guidelines for each task. However, we found that the annotators tend to never re-read the full **Instructions**, and to forget some of the important guidelines over time. Therefore, we have limited ourselves to 3 *golden rules* per task, that will be present all the time, directly in the annotation task screen.

T_{1a} **Golden Rules of the Claim Extraction:**

1. Please read the **Instructions** before making your first claim.
2. Make **simple true claims** based on the source paragraph that **make sense to be verified**.
3. If the source paragraph doesn't allow that or seems uninteresting, feel free to **Skip** it.

T_{1b} **Golden Rules of the Claim Mutation:**

1. Please read the **Instructions** before making your first mutation.
2. **! Only make mutated claims that make sense to be verified.**
3. Therefore, there is no need to use all 6 mutation types, 3 would suffice, even 1.

T_2 **Golden rules of the Claim Annotation**:

1. Before the first annotation, please, read the **Instructions**.
2. Pay attention to **the non-exclusivity of phenomena**, especially for the **Refute** annotations. *E.g. "a cinema is being built in Písek" does not refute "a gallery is being built in Písek".*
3. If the evidence alone is not sufficient, please provide the missing information as a *condition* of the annotation.

4.6.2 T2: The Action Flattening

After the first wave of annotation, that showed a significantly underwhelming usage of T_2 actions grouped in an **Actions** modal pop-up – especially the usage of the `NOT ENOUGH INFO` label, **Flag**s and *conditions* – we have re-thought the action toolbar and spread all the actions available onto a horizontal bar, each as a single button. If the action requires additional data, s. a. the **Flag** reason, it only asks for it in a "next step" modal window.

4.6.3 T2: User-Initiated Soft-Deletion

In addition to this, we have added the feature of *soft-deletes*. Thanks to the convenience of working with the Yii2 PHP Framework, we could simply constrain the system to only work with entity objects without the soft-deletion bit set to `1`, effectively augmenting each query above such an entity with "(...) `WHERE` (...) `AND deleted!=1`".

Since the Wave 2, each **Flag** was programmed to cause a temporary soft-delete of the flagged claim, to be re-considered by an administrator. This has shown to be a great synergy of the human-power of the annotators and admin's unconstrained system access. Admin got notified every time there was a claim containing a typo, a contradiction, or claim unrelated to the source paragraph, and did his best to fix the claim using his direct db access. In the meantime, the claim was inaccessible to the web user interface, and no annotation was wasted on invalid data.

Over the waves 2 and 3, we have received a total of **112** flags, effectively saving almost **4 hours** of compromised annotation, estimated using the average T_2 load (4.5.3) and 2 cross-annotations per claim. We have managed to recover **65** of the flagged claims to a state valid for the annotation task.

4.6.4 Spreading out the annotations

During the first wave, we have experienced a severe peak in annotators performance during the deadline (Figure 4.6). As relatable as that sounds, it did have a bad impact on the dataset quality.

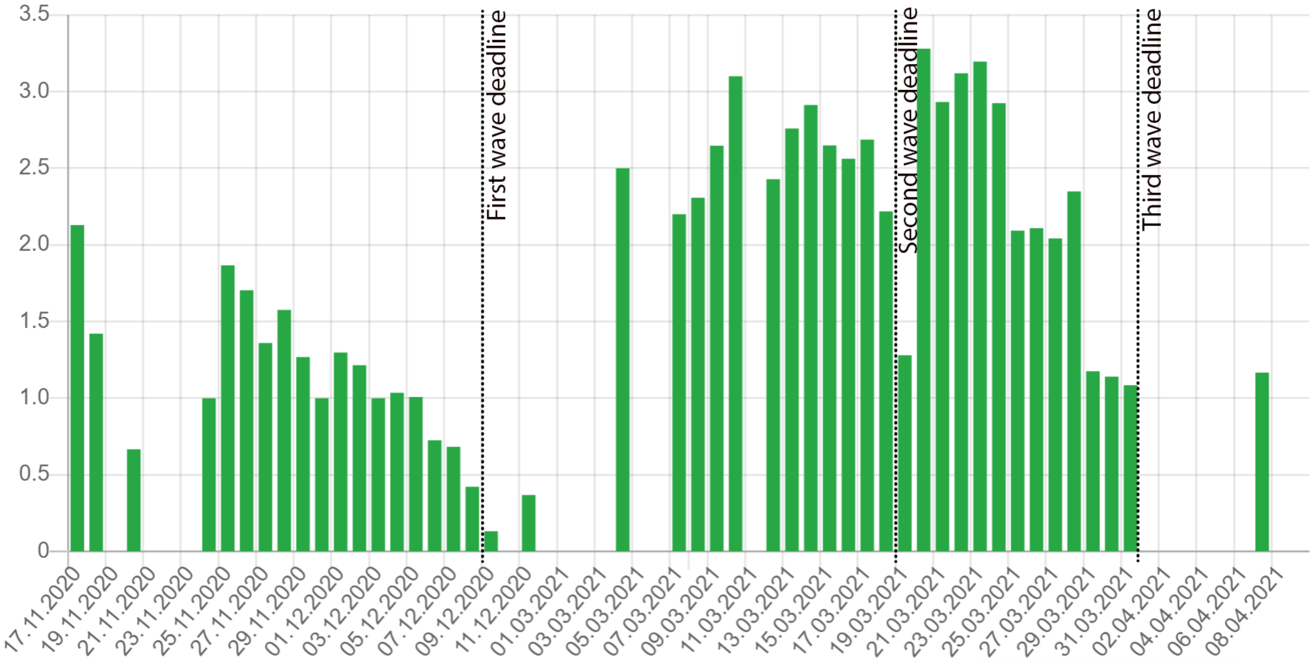


Figure 4.7: Average number of cross-annotations per claim by day. Only counting annotations within the current wave (+2 days tolerance – not showing the post-wave adjustments).

Due to the unbiased claim sampling in T_{2b}, the late claims were extremely punished, as they were absent for most of its instances. The Figure 4.7 shows this phenomenon – on December 8^{th} the average number of annotations per claim descended below 0.5, which is very unfortunate, as the Figure 4.6 shows this day to be the second most productive in the claim mutation.

Therefore, we have biased the T_2 claim sampling in the following ways:

1. m^c is assigned a random priority from $(0, 1)$
2. If m^c comes from the current wave of annotation, priority is incrementated by 1

3. If m^c has less then 1 non-oracle annotation[8], priority increments by 1

4. Finally, m^c with the highest priority is sent on output.

Fast implementation of this sampling using the SQL *subqueries* can be found in the attached `LabelController.php`.

In addition to this, we have **adjusted the required number of data-points per task**. From 5 T_{1a} claims, 15 T_{1b} mutations, 5 T_{2a} oracle annotations and 15 T_{2b} non-oracle annotations, we have switched to **3, 7, 7** and **35**, respectively (Figure 4.4), based on our stopwatch-per-task measurements and the findings from Figure 4.7, in which the 1^{st} *Wave* partition shows the need for an increase in cross-annotations.

Lastly, we have dedicated the time to hand-annotate ~**300 residual claims** after the last wave. These adjustments led to a significant improvement over the random baseline – for instance, after the first wave deadline, around 40% of all claims ended up with 0 annotations. By the time of the publication of this thesis, the amount of the 0-annotated claims decreased to only 9%, still counting the original 1^{st} wave set, subject to its post-annotations.

4.7 System Performance Remarks

We have been surprised by the robustness and traffic resistance of the resulting scheme. Dataset does not contain traces of blackouts or HTTP communication interrupts between Flask and Apache. We also consider the traffic load carried by our system during the wave deadlines (Fig 4.6) remarkable, given the size of the ČTK Archive and the complexity of the 4.3.4 algorithm. We attribute this to the full AJAX-initiated asynchronization of the costly operations and to the support we recieved from the Faculty of Electrical Engineering, namely Petr Benda, who provided us with a server with Intel Xeon E3 CPU and 132 GB of memory that runs both Flask and Apache apps to this date[9].

4.8 Annotation wrap-up

We have successfully conducted three experiments on human annotation for the *fact-checking using ČTK Archive* task, using a novel annotation platform of our own design, which is *live* on `https://fcheck.fel.cvut.cz`, or can be installed from its source[10] that is to be published under a license to be specified in `LICENSE.md`.

We thank to all FSS students who participated in our experiments for donating their time to support our endeavours with a total of **4,325** valid *Claim*, and **5,759** *Label* datapoints. According to their verbal feedback after the supplementary lectures, many of them enjoyed our cooperation[11], and the research partnership started with our project shall continue with other exciting future collaborations.

[8] Originally, we have tried to aim for 2 non-oracle annotations per claim, however, this was too punishing for the unannotated claims left from previous waves, as each new claim would be favoured for as long as it does not collect 2 annotations

[9] As of May 20^{th}

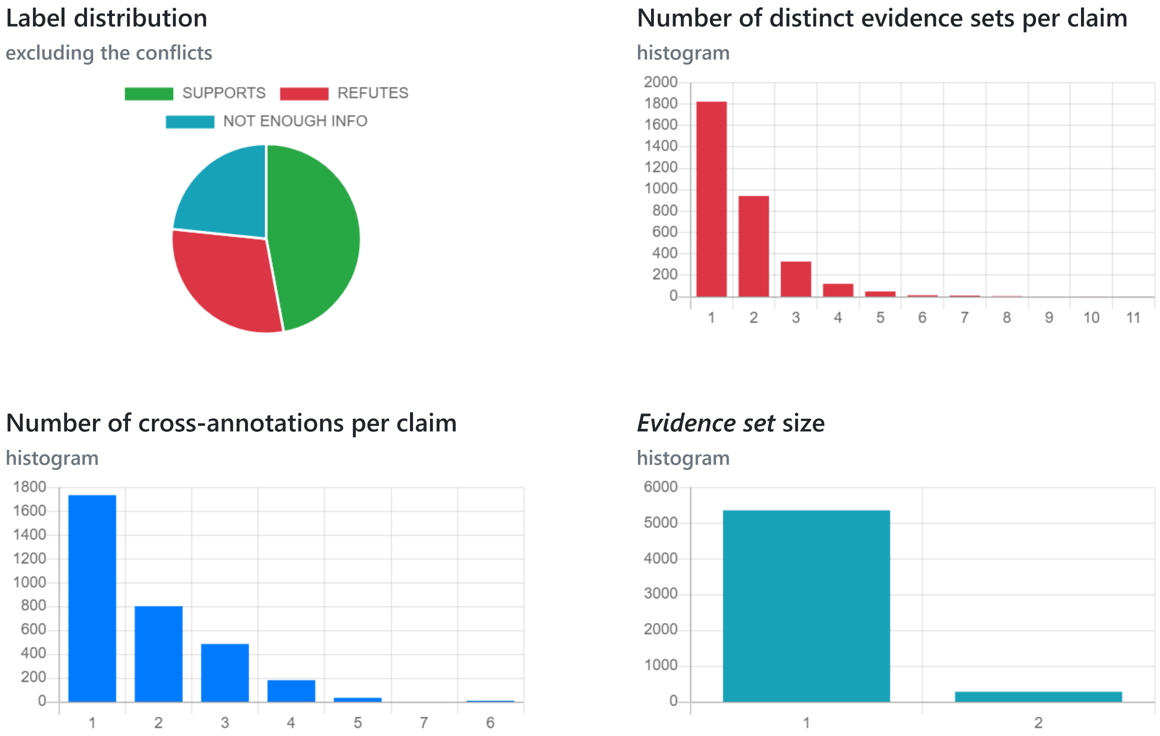
[10] `https://gitlab.fel.cvut.cz/factchecking/fcheck-anotations-platform`

[11] Even if, based on the textual inputs found in our database, at least one did not

Chapter 5
ČTK Dataset Analysis and Postprocessing

Through the methodology described in chapter 4, we have collected a set of raw claims and samples of their veracity labelling.

This chapter performs the exploratory analysis of the collected dataset, structured as described with Figure 4.1, and describes our methods of "flattening" it into a single text file that is easy to parse. Consecutively, we analyse the resulting dataset using several standard metrics and propose tools for its iterative refinement, ultimately leading to the current version of ČTK dataset, described and linked in 5.7.


Label distribution

excluding the conflicts

Number of distinct evidence sets per claim

histogram

Number of cross-annotations per claim

histogram

Evidence set size

histogram

Figure 5.1: Visualizations of properties of the collected dataset, extracted from our interactive dashboard (Section 5.1) attached to this thesis.

5.1 Live Dashboards

In order to provide our users with a comprehensible view into the resulting dataset and its properties, including the leaderboard of the most active annotators, we have implemented a dashboard of *live* data visualizations. For the live data aggregations, we have mostly used raw PHP and SQL, for interactive and æsthetically pleasing plots, we have used the Chart.js library.

We have decided to disclose the dashboard (with supplementary labels in Czech) to our reader, so as to accompany the texts to follow, and to provide a legible statistical insight into the data collected by the methods listed in Chapter 4.

It can be found at `https://fcheck.fel.cvut.cz/site/statistics` and logged into using the "`testuser`" SIDOS ID.

5.2 JSON Lines FCheck Export

For the purpose of *flattening* the FCheck database with all of its relations and metadata to a single concise text file used on an input for our end applications, we have constructed a JSONL API at `https://fcheck.fel.cvut.cz/label/export`.

It takes the following arguments through its HTTP GET parameters:

1. `shuffle` $\in \{0, 1\}$, defaults to 0
decides, whether the dataset should be shuffled using the MySQL's ...`ORDER BY rand()`

2. `evidenceFormat` $\in \{\texttt{text}, \texttt{ctkId}\}$, defaults to `ctkId`
if set to `text`, the full detokenized text of used ČTK paragraphs will be exported for evidence (preferred for NLI), otherwise, only the underscore-separated article and paragraph id will be given, s.a. `T201810060771501_2` (preferred for DR)

3. `summer` $\in \{0, 1\}$, defaults to 0
decides, whether the first wave of annotations should be excluded from the export, by default it is sorted by the source paragraph

4. `fever` $\in \{0, 1\}$, defaults to 0
switches to the FEVER-like output format (3.1), to ease the usage of FCheck data for experiments implemented to run with FEVER CS – setting to 1 disables the other options for legacy reasons

5.2.1 ČTK dataset formats

While much like [Thorne et al., 2018a] we use the JSONL file type, which we deem appropriate for the fact verification datasets, we propose an alternate format for flattening the data points than that from Figure 3.1. In our case, we argue to suppress all the auxiliary information except the *Claim* `id` to refer back to the 4.1 representation of data, following the KISS[1] and YAGNI[2] [Jeffries et al., 2001] design principles.

While this interpretation of the *labeled-claim* datapoints is our current default, we also enable switching back to the FEVER format illustrated in 3.1, using the `fever` flag for backwards compatibility. This is particularly useful for reusing the model training procedures written for FEVER CS.

We demonstrate the two types of *Evidence* representation in Figures 5.2 and 5.3 – `text` is meant for the training and validation of the NLI models (Chapter 7), whereas `ctkId` suits the Retrieval tasks. For the completeness, the `ctkId` consists of the ČTK Archive identifier and the paragraph 1-based index in the archived article, separated by underscore. Index 0 is reserved for the article headline as it may also be used for both tasks.

[1] "Keep It Simple, Stupid!"

[2] "You Ain't Gonna Need It."

```
{
  "id": 2500,
  "label": "SUPPORTS",
  "claim": "S Dejvickým divadlem spolupracoval Petr Zelenka.",
  "evidence": [
    ["T201111230392601_9"],
    ["T200708140695001_2"]
  ]
}
```

Figure 5.2: Example ČTK `SUPPORTS` annotation with two possible evidence sets, each composed of one ČTK paragraph, using the `ctkId` evidence format.

```
{
  "id": 2500,
  "label": "SUPPORTS",
  "claim": "S Dejvickým divadlem spolupracoval Petr Zelenka.",
  "evidence": [
    ["Petr Zelenka vystudoval scenáristiku a dramaturgii na FAMU (…) V
   ↪ roce 2001 napsal pro Dejvické divadlo Příběhy obyčejného šílenství
   ↪ , za které získal Cenu Alfréda Radoka (…)"],
    ["(…) kterou Zelenka později režíroval i jako stejnojmenný film. V
   ↪ roce 2005 uvedl na dejvické scéně svou další hru Teremin. V
   ↪ současné době natáčí Zelenka osobitou verzi inscenace Dejvického
   ↪ divadla Karamazovi."]
  ]
}
```

Figure 5.3: The same example as 5.2 using the `text` evidence format, paragraphs were truncated using (…)

```
{
  "label": "SUPPORTS",
  "claim": "S Dejvickým divadlem spolupracoval Petr Zelenka.",
  "context": [
    "(…) Petr Zelenka (…) napsal pro Dejvické divadlo (…)"
  ]
}
```

```
{
  "label": "SUPPORTS",
  "claim": "S Dejvickým divadlem spolupracoval Petr Zelenka.",
  "context": [
    "(…) kterou Zelenka (…) 2005 uvedl na dejvické scéně (…) "
  ]
}
```

Figure 5.4: The same example as 5.3 using the `nli` evidence format, paragraphs were truncated using (…). Note that in this format we have 2 datapoints - one for each evidence set.

Additionally, we introduce the ČTK `nli` format that is appropriate for training and testing the Natural Language Inference models in the Figure 5.4, note that this format produces a different number of datapoints, one for every evidence set, listed as `context`.

5.3 Cross-annotations

In FEVER annotation labeling task WF2 [Thorne et al., 2018a], the annotators were advised to spend not more than 2-3 minutes to find as many of the evidence sets as possible in the given dictionary (and even using a direct Wiki access), so that the dataset can later be considered *exhaustive*, i.e., to boost its *evidence recall*, which was later computed to be **72.36%**.

With our ČTK Archive corpus, this is unrealistic, as it commonly contains an inconceivable number of copies for a single ground truth[3]. So is the number of paragraphs in the *mutated claim*'s knowledge scope, typically close to (see T_{1b} and 4.3.4 for reference) $\max_{m^c}|knowledge(m^c) \cup knowledge(p) \cup \{p\}| = 17$. Therefore, we proposed a different scheme: we advised every annotator to spend 2-3 minutes finding a *reasonable* number of distinct evidence-sets, w.r.t. the time needed for a good reading comprehension. Furthermore, we randomly shuffled the set of all *knowledge scope* documents using PHP's `shuffle`[4] before the start of every T_2 annotation.

As the annotators typically skim through the knowledge headlines in a top-first order, this made it difficult for two annotators to arrive to the same set of evidence-sets. To exploit that, we collected multiple *cross-annotations* for each claim – their distribution is best visualized with the histograms in Figure 5.1. Finally, as a subroutine of our export tool 5.2, we merge the evidence of all the cross-annotations for a given claim together, to achieve the highest possible *recall*.

5.4 Inter-Annotator Agreement

A desirable byproduct of the cross-annotation-driven approach above are the large resulting groups of k-way *labeled* claims. I.e., the claims that were assigned exactly k independent labels from $\{\mathtt{SUPPORTS}, \mathtt{REFUTES}, \mathtt{NEI}\}$ by different annotators.

To measure the agreement using the most straightforward implementations of the measures enumerated in Table 5.1, we first conclude two *pairwise* agreement experiments, first using the average 0/1-agreement measure (listed as the %-aggreement), then the Cohen's κ [Cohen, 1960], which is the standard for bipartite agreement. By *pairwise experiment*, we mean an exp't concluded using the enumeration of all the pairs of labels of every (≥ 2)-way labeled claim on its input.

Secondly, we examine each k-way annotated partition of claims using the Fleiss' κ measure introduced in [Fleiss, 1971], which is the standard for the k-way inter-annotator aggreement. We list its results on the most significant (> 2)-way-annotated partitions of our dataset, along with the share of the partition in the whole dataset, denoted as the *Claim-Coverage*.

[3] Think, the proposition "Miloš Zeman is the Czech president", which can be found in every "(...), said the Czech president Miloš Zeman."

[4] Which internally uses the Mersenne Twister pseudorandom number generator, for the completeness

Metric	Value	Agreement[5]	Claim-Coverage[6]
Pairwise percent agreement	**74%**		55.9%
Pairwise Cohen's κ	**0.58**	*moderate*	55.9%
3-way Fleiss' κ	**0.57**	*moderate*	19.3%
4-way Fleiss' κ	**0.63**	*substantial*	7.5%
4-way Krippendorff's α	**0.63**	*substantial*	7.5%
5-way Fleiss' κ	**0.61**	*substantial*	1.7%

Table 5.1: Inter-annotation agreement metrics of the ČTK v2.1 dataset, excluding the *conditional annotations*

Experimentally, we have also calculated the Krippendorff's α from [Krippendorff, 2013], which yielded the same results as the Fleiss' κ up to 2 decimal spots. Krippendorff's α should be appropriate for the agreement experiments with missing data, measuring the within- and between-unit error. We encourage a further experimentation using Krippendorff's α and the entire ČTK dataset augmented by the annotator identifiers in the future.

The measurements can be replicated using the attached `agreement.py` Python module and the cross annotation May'21 snapshot in `cross_annotations.csv`. We consider the results promising with respect to the complexity of the T_2 task and the ČTK corpus. To put in context, [Thorne et al., 2018a] achieved a 5-way Fleiss' kappa of **0.68** using a simpler ENWiki dataset, dictionary structure and longer total time per annotator, affecting the *learning curve*. [Binau and Schulte, 2020] demonstrated the importance of this factor by achieving **0.75** κ-score through only using two expert-level annotators – themselves. We conclude that the dataset is usable for the fact-verification task, which we will demonstrate on its NLI subroutine (Chapter 7).

5.4.1 Annotation Cleaning

We have dedicated a significant amount of time to manually traverse *every* disagreeing pair of annotations, to see if one or both of them violate the annotation guidelines. The idea was that this should be a common case for the conflicting annotations, as the ČTK News Archive corpus does not commonly contain a conflicting pair of paragraphs except for the case of *temporal reasoning* shown in Chapter 4. In the same chapter, we have resolved this case using the *claim timestamps*, that always favour the latest knowledge published up to the given date.

Indeed, after separating out the incorrectly formed annotations using our *soft-deletion* mechanisms introduced in the section 4.6.3, we have been able to resolve every conflict, ultimately achieving a *full agreement* between the annotations. However, the metrics listed in Table 5.1 do *not* exclude the soft-deleted labels, so as to provide a better insight into the reliability of the data *without* the conflict.

5.5 Common annotation problems

After removing hundreds of ineligible annotations in 5.4.1, we would like to mention several archetypes of their underlying problems. Their avoidance should be put into cosideration when designing similar annotation experiments in the future.

[5] The verbal interpretation is provided for reader's convenience and follows the interpretation tables of [Landis and Koch, 1977] which are mainly orientational and by no means universally accepted.

[6] The percentage of labeled claims eligible for this experiment out of the entire set.

1. **Exclusion misassumption** – by far the most prevalent type of misclassification: the annotator uses a ground truth independent of the claim as a `REFUTES` evidence. E.g., *"Postoloprty opened a new cinema"* ***REFUTES*** *"Postoloprty opened a new museum".*

   While on the first sight, this might seem like a sound disproof, there is no textual entailment between the claims nor their negations. We attribute this error to confusing the T_2 with a *reading-comprehension*[7] task common for the field of humanities.

   We have reduced the frequency of this misclassification by introducing a *golden rule* (Section 4.6.1) for it, keeping it on annotator's mind at all times

2. **Mutation vagueness** – *Claim* fault. Mutation generalizes out an integral part of the original claim, typically the named entities. E.g., $m^c =$ "The convoy is 200 metres long"

3. **Temporal reasoning** – an inherent problem of the journalistic datasets – an annotater submits a dated evidence paragraph that contradicts the latest news w.r.t. $timestamp(m^c)$

4. **NEI "shyness"** – "Pandas are endangered." was used once for `SUPPORTING` and once for `REFUTING` the claim "Koalas are endangered.", zero times as `NEI`. This, among other examples, shows that our annotators often preferred the *definite* labels, even where `NEI` is appropriate, which might justify its underrepresentation shown in Figure 5.1.

   We tried to address this introducing the *conditional* annotations (Section 4.4.2).

For the completness, we attach the raw file `archetypy.docx`[8] in Czech, naming multiple examples from ČTK v1 dataset for each of the archetypes above.

5.6 Legacy version notes on the ČTK dataset

As there were several different export snapshots of our data used for the experiments in Chapter 7 and the work of [Rýpar, 2021], we include version notes for the major dataset versions to refer back to:

- **ČTK dataset v1** – December 2020

  Legacy dataset published in the FEVER format (3.1), featured the first wave of ~**950** annotations, highly experimental, significantly helped to reveal the data faults described in the Section 5.5.

- **ČTK dataset v2** – April 2020

  Cleaned (5.4.1). Contains the first snapshot of data from all three waves, ignoring *conditional annotations* and conflicts. Follows the label distribution from Figure 5.1 using a *stratified* train-dev-test split generated through two iterations of scikit-learn's `train_test_split`, each with a fixed *random seed* and a test size of 0.2

- **ČTK dataset v2csv**

  Generated in parallel as a part of Jan Drchal's research from the May snapshot of FCheck db. It attempts to minimize the *document leakage* 3.2.5 by sorting the claims by their source paragraph before the train-dev-test split. Ignores the evidence grouping (4.1), however, yields encouraging results for NLI.

[7] *"Does the article tell us that Postoloprty opened a museum? Highlight the relevant information."*

[8] `http://bertik.net/archetypy.docx`

- **ČTK dataset v*nli**

  Augmented using the techniques introduced in Section 7.3, formatted as a JSONLines of 5.4 datapoints, using the same db snapshot as the version given instead of * symbol

5.7 Resulting Dataset

Finally, we publish[9] our final version of the ČTK dataset collected using the platform described in Chapter 4 and cleaned using the scheme introduced in Section 5.4.1. We are attaching the dataset in two different formats, that is, the **ČTK v2.1**, which is exported into a FEVER-like JSONL (3.1) for the Document Retrieval task, currently being used by [Rýpar, 2021] and the augmented (7.3) **ČTK v2.1nli** using the standard NLI format (5.4), that will be used in Chapter 7, stored both in *label-uniform* and *stratified*[10] train-dev-test split.

	ČTK v2.1			**ČTK v2.1nli**			**ČTK v2.1nli stratified**		
	SUPPORTS	REFUTES	NEI	SUPPORTS	REFUTES	NEI	SUPPORTS	REFUTES	NEI
train	1,132	519	473	2,052	792	1311	1,775	900	1255
dev	100	100	100	167	167	167	266	134	188
test	200	200	200	333	333	333	511	258	361

Table 5.2: Label distribution in our ČTK v2.1 dataset (with forced label uniformity in the validation sets to remove advantage for heavily biased predictors) and in our ČTK v2.1nli uniform and stratified splits

The data collection and refinement experiments can be reproduced using the methodology described by the Chapter 4, the exports and formatting are described in the previous sections of this chapter and can be re-instantiated using our dataset cleaning[11] web interface and the *flattening* API, disclosed in 5.2.

The inter-annotation measures collected in 5.3 suggest that we got our hands on a sufficiently reliable, and certainly a very exciting testbed for the *fact-verification* solutions working within the largely unexplored framework of the news-archive corpora...

[9] http://bertik.net/ctk

[10] Maintaining the same label distribution in all datasets.

[11] https://fcheck.fel.cvut.cz/label/clean

Part II

Natural Language Inference in Czech

Chapter 6
The AIC/FactCheck Context

Now that we have collected and validated two training, development and testing datasets in chapers 2 and 5, let us spend a short chapter on how this data is being practically used to build a fact verifier under the appropriate knowledge base.

We will explore the works of [Gažo, 2021, Dědková, 2021] and [Rýpar, 2021], our colleagues from the research group FactCheck at AIC, to establish how the *document retrieval* subproblem is being solved, and outline the format and characteristics of its output. This will be referred to in Chapter 7 on Natural Language Inference, which takes a claim and a set of evidence on its input and outputs a veracity verdict from {`SUPPORTS`, `REFUTES`, `NOT ENOUGH INFO`}.

We will also briefly discuss the "end user" application demonstrations we have prepared for the fact-checking task and its subroutines in the past, to specify how the outputs of the following chapters shall be used in practice.

6.1 Document retrieval task

During the summer semester 2021, we have subdivided the *fact-checking pipeline* tasks from Figure 1.2 among the members of our team, as described in 1.2.1. While our work is held accountable for the software engineering, experiment design and the validation schemes neccessary for establishing the Czech fact-checking datasets, the work of [Rýpar, 2021][1] takes their snapshots and uses them to train and validate the Document Retrieval models.

The Document Retrieval model takes a textual claim on input and outputs a set of Documents (e.g. ČTK paragraphs or Wikipedia abstracts) from a fixed domain – the *knowledge base*. We refer to its result as to the *evidence set*, as it shadows the concept of evidence sets introduced in Figure 3.2 and present in our datasets (though in 6.1.1, we will argue that we only need a reasonable-sized *superset* of the dataset-like evidence set).

To follow up, we dedicate the current Part of our thesis to train a model which, given such an evidence set on its input along with a textual claim, outputs a veracity label to conclude the fact-checking verdict. Therefore, we find it vital for the text of our thesis, to include a brief look into the previous task on the pipeline and see the models that, in the end-applications will be feeding their output into our Natural Language Inference model (Chapter 7) and examine its form and reliability.

[1] The works of [Gažo, 2021] and [Dědková, 2021] were postponed to a later deadline, partly due to the distance learning during the Czech COVID-19 surge, and did not yet deliver a solution to experiment with.

6.1.1 Recall Over Precision

The standard metrics for the retrieval task are the *Precision* and the *Recall*. Loosely speaking, precision characterizes the overall *quality* of the results as the percentage of relevant results in the entire output ($precision = \frac{true\ positives}{true\ positives+false\ positives}$) whereas the recall expresses the *quantity* of the relevant results, as the percentage of the *retrieved* relevant results in the set of *all* relevant results w.r.t. dataset ($recall = \frac{tp}{tp+false\ negatives}$). Their *harmonic mean* is called the F_1-score, and is commonly used for measuring the quality of the Retrieval models, as it punishes the unwanted *tradeoffs* between *precision* and *recall*.

For us, this is not the case – due to the *self-attention* mechanism described in 1.3, we presume the NLI models to be rather forgiving to the precision faults, i.e. to be able to find the conclusive part of evidence even in a rather long input. Therefore, we use the *recall* as our default benchmark for the Document Retrieval models trained by [Rýpar, 2021] and [Pitr, 2020], as even the task of scaling down the entire ČTK db from 10^8 paragraphs to, say, a set of 20, that are guaranteed to contain an *evidence set* yields an admissible input for the NLI models discussed in Chapter 7.

6.1.2 Internal State of the Art

FEVER CS dev set

model	R@1	R@2	R@5	R@10	R@20
DRQA	38.99	51.68	63.74	69.85	74.66
Anserini BM25 finetuned	39.30	49.94	61.13	67.78	73.07
mBERT BFS+ICT	**61.48**	**75.62**	**87.34**	**91.88**	**94.40**
ColBERT_128dim (FEVER CS)	51.64	62.84	71.32	75.22	78.28
ColBERT_128dim (ČTK + FEVER CS)	43.71	54.59	64.84	70.87	75.28
ColBERT_64dim (ČTK + FEVER CS)	41.31	51.53	61.37	67.19	72.02

ČTK v2.1 test set

model	R@1	R@2	R@5	R@10	R@20
DRQA	12.75	19.25	25.50	31.00	35.50
Anserini BM25 finetuned	15.75	22.00	29.25	33.75	39.75
ColBERT (FEVER CS + ČTK)	**19.50**	**27.25**	**35.25**	**40.00**	**46.00**
mBERT BFS+ICT (FEVER CS + ČTK)	1.00	2.25	5.00	8.25	12.25

Table 6.1: Percent-recall for a fixed output size of k paragraphs, measured using the FEVER CS and ČTK datasets. Reprinted from [Rýpar, 2021], bold values signify the best result.

In Table 6.1, we show the measurements of the *recall*[2] of most significant retrieval models trained by [Rýpar, 2021] and [Pitr, 2020] from AIC, using the computing power of the RCI Cluster. Even though the numerical models, such as DrQA (*tf–idf*) and the Anserini implementation of Okapi BM25 (*bag-of-words*) methods set a strong baseline to validate the Transformer training, they are ultimately surpassed by the neural BERT-like models.

AIC CTU's internal sota for Document retrieval is a *two-tower retrieval model* [Chang et al., 2020] based on mBERT [Devlin et al., 2019], pretrained on Czech Wikipedia corpus

[2] For the less task-relevant (but more standard) F_1 measure, see the work of [Rýpar, 2021], linked in the Bibliography.

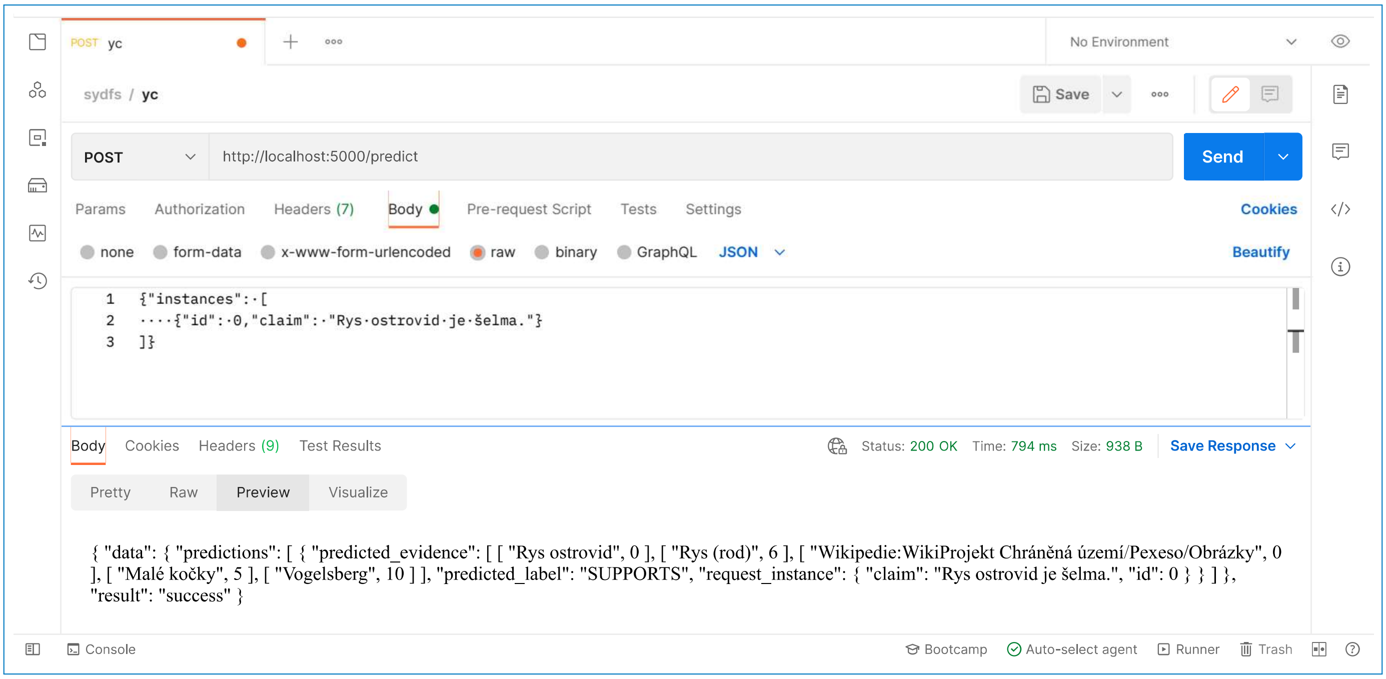
for using the Body First Selection and Inverse Cloze tasks, which achieves a **91.88%** test recall on the FEVER CS data for a fixed output size of 10 documents.

For the ČTK Paragraph retrieval, Rýpar proposes a ColBERT [Khattab and Zaharia, 2020] model trained using $(claim, evidence\ document, non\text{-}evidence\ document)$ triples from both the FEVER CS and the ČTK dataset, which has a **40%** recall at 10 output documents. The ČTK dataset is to be further examined for faults, as the mBERT recall is shockingly low on it, given it was this very model to compute significant part of the *knowledge scopes* (4.3.4).

6.2 Production

6.2.1 FEVER CS Baseline (Sep 2020)

In the *Software or Research Project* precedent to this thesis, we have released a baseline FLASK API that performs the end-to-end fact verification (as shown in 1.2) of any given textual claim using the CS Wikipedia knowledge base. It follows the format for the FEVER shared task submissions [Thorne et al., 2018b] and it is fully containerized to run using a simple docker or singularity run command. It can be run directly from the DockerHub `ullriher/fever-cs-baseline` repository, or built from the source[3]. The published *baseline* system uses dated models – the DrQA for Document Retrieval and a *Decomposable Attention model* for the Natural Language Inference – which are to be updated with the solutions proposed by of our current research (Chapter 8). We include it for a tangible example of product utilizing our theoretical findings and show an example server interaction in Figure 6.1.



POST http://localhost:5000/predict

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{"instances": [
    {"id": 0,"claim": "Rys ostrovid je šelma."}
]}
```

Status: 200 OK Time: 794 ms Size: 938 B

```
{ "data": { "predictions": [ { "predicted_evidence": [ [ "Rys ostrovid", 0 ], [ "Rys (rod)", 6 ], [ "Wikipedie:WikiProjekt Chráněná území/Pexeso/Obrázky", 0
], [ "Malé kočky", 5 ], [ "Vogelsberg", 10 ] ], "predicted_label": "SUPPORTS", "request_instance": { "claim": "Rys ostrovid je šelma.", "id": 0 } } ] },
"result": "success" }
```

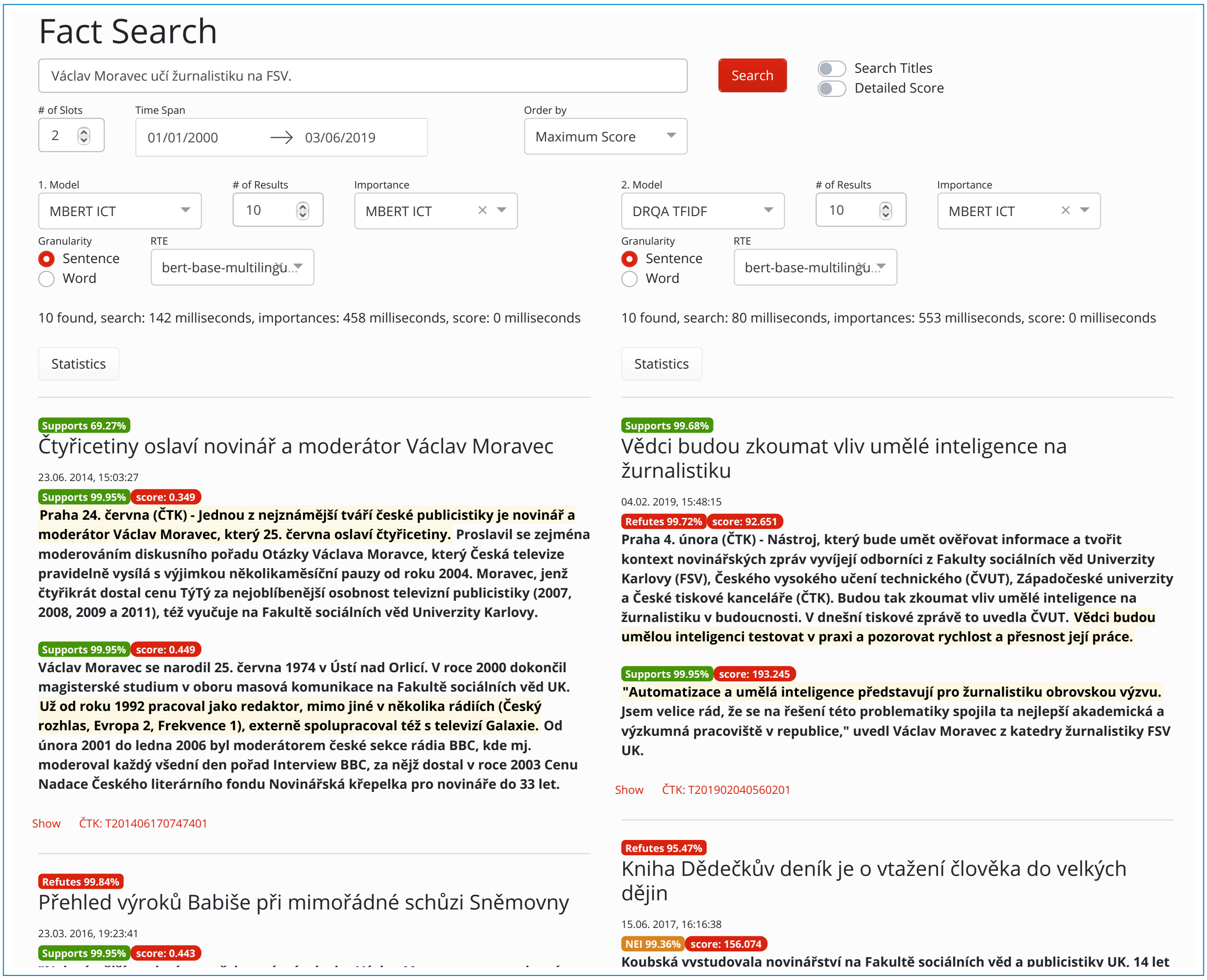
Figure 6.1: Our fever-cs-baseline API, accessed through Postman

[3] `https://github.com/aic-factcheck/fever-cs-dataset`

6.2.2 Fact Search

Through the course of the last year, our team supervisor Jan Drchal has produced a plethora of demonstrative production-like interfaces for our meetings with the FSS and TAČR stakeholders. The most notable interface is the Fact Search web console (Figure 6.2) that emulates the outputs of the Document Retrieval task for an arbitrary claim, knowledge base and DR model.

It computes the single-paragraph inference label using a legacy model for RTE (also known as NLI) for each retrieved document, to show the NLI use case. It also illustrates the Document Retrieval task referred to in 6.1 and gives a real-world example of an input data for our next chapter…



Fact Search

Václav Moravec učí žurnalistiku na FSV.

# of Slots: 2
Time Span: 01/01/2000 → 03/06/2019
Order by: Maximum Score

1. Model: MBERT ICT
# of Results: 10
Importance: MBERT ICT
Granularity: ● Sentence ○ Word
RTE: bert-base-multilingu…

10 found, search: 142 milliseconds, importances: 458 milliseconds, score: 0 milliseconds

Supports 69.27%

Čtyřicetiny oslaví novinář a moderátor Václav Moravec

23.06. 2014, 15:03:27

Supports 99.95% score: 0.349

Praha 24. června (ČTK) - Jednou z nejznámější tváří české publicistiky je novinář a moderátor Václav Moravec, který 25. června oslaví čtyřicetiny. Proslavil se zejména moderováním diskusního pořadu Otázky Václava Moravce, který Česká televize pravidelně vysílá s výjimkou několikaměsíční pauzy od roku 2004. Moravec, jenž čtyřikrát dostal cenu TýTý za nejoblíbenější osobnost televizní publicistiky (2007, 2008, 2009 a 2011), též vyučuje na Fakultě sociálních věd Univerzity Karlovy.

Supports 99.95% score: 0.449

Václav Moravec se narodil 25. června 1974 v Ústí nad Orlicí. V roce 2000 dokončil magisterské studium v oboru masová komunikace na Fakultě sociálních věd UK. Už od roku 1992 pracoval jako redaktor, mimo jiné v několika rádiích (Český rozhlas, Evropa 2, Frekvence 1), externě spolupracoval též s televizí Galaxie. Od února 2001 do ledna 2006 byl moderátorem české sekce rádia BBC, kde mj. moderoval každý všední den pořad Interview BBC, za nějž dostal v roce 2003 Cenu Nadace Českého literárního fondu Novinářská křepelka pro novináře do 33 let.

Show ČTK: T201406170747401

Refutes 99.84%

Přehled výroků Babiše při mimořádné schůzi Sněmovny

23.03. 2016, 19:23:41

Supports 99.95% score: 0.443

2. Model: DRQA TFIDF
# of Results: 10
Importance: MBERT ICT
Granularity: ● Sentence ○ Word
RTE: bert-base-multilingu…

10 found, search: 80 milliseconds, importances: 553 milliseconds, score: 0 milliseconds

Supports 99.68%

Vědci budou zkoumat vliv umělé inteligence na žurnalistiku

04.02. 2019, 15:48:15

Refutes 99.72% score: 92.651

Praha 4. února (ČTK) - Nástroj, který bude umět ověřovat informace a tvořit kontext novinářských zpráv vyvíjejí odborníci z Fakulty sociálních věd Univerzity Karlovy (FSV), Českého vysokého učení technického (ČVUT), Západočeské univerzity a České tiskové kanceláře (ČTK). Budou tak zkoumat vliv umělé inteligence na žurnalistiku v budoucnosti. V dnešní tiskové zprávě to uvedla ČVUT. Vědci budou umělou inteligenci testovat v praxi a pozorovat rychlost a přesnost její práce.

Supports 99.95% score: 193.245

"Automatizace a umělá inteligence představují pro žurnalistiku obrovskou výzvu. Jsem velice rád, že se na řešení této problematiky spojila ta nejlepší akademická a výzkumná pracoviště v republice," uvedl Václav Moravec z katedry žurnalistiky FSV UK.

Show ČTK: T201902040560201

Refutes 95.47%

Kniha Dědečkův deník je o vtažení člověka do velkých dějin

15.06. 2017, 16:16:38

NEI 99.36% score: 156.074

Koubská vystudovala novinářství na Fakultě sociálních věd a publicistiky UK. 14 let

Figure 6.2: Fact Search demo, authored by Jan Drchal, code at [Drchal and Ullrich, 2020]

Chapter 7
Natural Language Inference

In chapter 6, we have discussed the methods chosen by the FCheck team to retrieve a set of evidence relevant to the given claim. In the following chapter, we will proceed to show how to use these sets of evidence to infer whether the claim is provable or refutable.

This task is widely known as the *Natural Language Inference* (NLI), previously known as the *Recognizing Textual Entailment* (RTE). Whereas the RTE classification is bipartite (*entailed*, *no entailment*), the standard NLI classification is tripartite (*entailed*, *negation entailed*, *no entailment*) [Chatzikyriakidis et al., 2017].

7.1 Task definition

Given a textual claim c_i and its set of evidence E_{c_i} from the knowledge base, give a veracity label

$$h(c_i, E_{c_i}) = y$$

where $y \in \{\texttt{SUPPORTS}, \texttt{REFUTES}, \texttt{NOT ENOUGH INFO}\}$

For a practical instance, given a blinded datapoint formatted as in 5.4, give the corresponding `label` given the `claim` and a `context`.

7.2 Related work

We have examined the following NLI datasets in English, and their respective state-of-the-art classifiers, largerly based on transformer models resemblant to BERT [Devlin et al., 2019]

- **Stanford NLI Corpus** (**SNLI**) [Bowman et al., 2015]: “A large annotated corpus for learning natural language inference” – a long-term standard benchmark for the task of natural language inference. Corpus of ~570,000 human-written English sentence pairs manually labeled for balanced classification as `entailment`, `contradiction` or `neutral`.

  The state-of-the-art classifier as of May 2021 is EFL [Wang et al., 2021], which reaches 93.1% accuracy on the testing set. It uses a few-shot learning of RoBERTa [Liu et al., 2019] on the specific NLI classes.

- **Multi-Genre Natural Language Inference** (**MultiNLI**) [Williams et al., 2018] was collected for the RepEval shared task. It is modeled after SNLI and distributed in the same format. It contains ~433,000 sentence pairs and covers various genres of spoken and written English, such as FICTION extracted from project Gutenberg[1], TRAVEL from Berlitz travel guides, etc...

As of May 2021, The highest accuracy (92.2% in MATCHED, 91.7% in MISMATCHED) was reached by Google's T5-11B[2] [Raffel et al., 2019] through *transfer learning*, i. e. fine-tuning a large model pretrained on a data-rich task to the specific downstream task of NLI.

- **Adversarial NLI** (**ANLI**) [Nie et al., 2019b] - human-and-model-in-the-loop dataset, consisting of three rounds of increasing complexity and difficulty (A1, A2, A3), that include explanations provided by annotators. The total size of all sets is about 170K sentence pairs.

  The state-of-the-art solver InfoBERT [Wang et al., 2020] applies a further adversarial training to the RoBERTa model to achieve 75% accuracy on the A1 test set and 58.3% overall, using all the samples of A1, A2, A3 test sets combined.

- **FEVER for NLI** is a simple conversion of the FEVER dataset from its original format to the $(query, context)$ pairs, made as a byproduct of the UNC classifier [Nie et al., 2019a] from the FEVER shared task.

  This specific classifier was taught using NSMN[3] augmented by a "relatedness" score and ontological knowledge from WordNet, and achieved 68.16% label accuracy.

7.2.1 Slavonic Language Models

As nearly every solution examined in the Section 7.2 relies on the transfer learning, fine-tuning a large Transformer (1.3) language model to learn the predictions on a *down-stream* task, let this be the strategy we employ for the preliminary entailment experiments as well.

First of all, let us examine the models that may already "speak" Czech:

- **Multilingual BERT** (**mBERT**) – is, basically, a variation of Google's famed BERT$_{\text{BASE}}$ model [Devlin et al., 2019] for multiple languages, trained on the *Masked Language Modeling* (*MLM*) and *Next Sentence Prediction* (*NSP*) tasks using 104 localizations of Wikipedia. In our team, it has already been used for the *knowledge scope* computations (4.3.4), as well as for the Document Retrieval task by [Rýpar, 2021, Pitr, 2020] (6.1) towards encouraging results

- **SlavicBERT** [Arkhipov et al., 2019] – similar to mBERT, trained on joint Bulgarian, Czech, Polish and Russian corpora

- **HerBERT Ullrich** – haha, nothing here. Just testing your *attention* (1.3)[4]

- **XLM-RoBERTa** [Conneau et al., 2019] is the crosslingual version of RoBERTa, trained solely on the *MLM* task on a corpus significantly larger than of Wikipedia – the cleaned CommonCrawl data it is trained on comes at 2.5TB of storage-size. As of May 2021, the RoBERTa derivates achieve the sota performance in many NLI benchmarks[5]

[1] `https://gutenberg.org`

[2] Stands for a **T**ext-**t**o-**T**ext **T**ransfer **T**ransformer with **11 B**illions of parameters.

[3] Neural Semantic Matching Network

[4] *Ba dum tss.*

[5] See at `https://paperswithcode.com/task/natural-language-inference/latest`

- **CZERT** [Sido et al., 2021] is a recent arrival to the BERT family – a couple of monolingual Czech models that are trained using 340K sentences from the Czech Wikipedia, Czech National Corpus and crawled Czech news. The models are based on BERT and ALBERT [Lan et al., 2019] and are trained with random initialization on the *MLM+NSP* and *MLM+Sentence Order Prediction* tasks, respectively.

7.3 Modified ČTK dataset

On top of the ČTK dataset (Section 5.7), we propose the following simple methods of augmentation for the NLI task, using the auxiliary data collected in Chapter 2.

To emulate a set of evidence for the `NOT ENOUGH INFO` annotations in task 7.1, one can simply sample paragraphs from the knowledge scope of this claim. We propose to sample multiple different evidence sets for a single `NEI` claim in order to balance the dataset.

In 4.4.2, we have introduced the concept of *conditional labeling.* In terms of natural language inference, the labels with nonempty `condition` can be included twice:

1. As `NOT ENOUGH INFO` annotations
2. As `SUPPORTS` or `REFUTES` annotations, if we consider larger evidence sets, augmented with the knowledge listed in `condition`

This behaviours have been added to our dataset export API (5.2) using the HTTP GET activation parameters `simulateNei=1` and `condition=double`, respectively.

7.4 NLI Experiments

In the following section, we will conclude a set of preliminary Natural Language Inference experiments, largely relying on the AIC FactCheck's internal set of modules for model training and evaluation by [Drchal and Ullrich, 2020], that has shown promising preliminary results after the first wave of annotations.

The BERT-like models are handled with ease using the Huggingface `transformers` [Wolf et al., 2019] and the sBERT `sentence_transformers` [Reimers and Gurevych, 2019] libraries. We would like to thank the authors of all the aforementioned software for making it easy to obtain relevant results without having to delve too deep into the underlying compatibility challenges.

7.4.1 Data Consistency Remarks

As the experiments from this chapter were to be run simultaneously with the dataset collection and evaluation from the Chapters 4 and 5, naturally, we have run into a *race condition.* That is, at a certain point we had to fix a single legacy dataset version to be used for all further runs of our experiments and to proceed with the production dataset refinement in separation from the NLI application.

The dataset we fixed for this task is the ČTK v2 CSV (5.6) exported in early May 2021, that does not yet include all the refinements described in Chapter 5. However, it features all the datapoints from the first three waves, similarly to ČTK v2.1. For the completeness, we attach the JSONL reprint of the data in our dataset cloud storage[6], even though in

[6] `http://bertik.net/ctk`

practice it is being generated on-demand from the CSV db dump using a fixed *seed of randomness*.

As we will be using the *stratified* split of our dataset, de facto disabling the *accuracy* metric, we will be comparing our models based on their the F-score, which is a standard benchmark for the NLI task [Poliak, 2020].

7.4.2 Experiments on Sentence_transformers Models

Using a set of ad–hoc Jupyter notebooks[7] powered by the sentence_transformers and the shared codebase of the AIC/FactCheck team, we have downloaded the pretrained SlavicBERT and mBERT models in their *cased* defaults, provided by the DeepPavlov [DeepPavlov, 2021]. Furthermore, we have acquired an XLM-RoBERTa model, fine-tuned on an NLI-related *SQuAD2* [Rajpurkar et al., 2016] *down-stream task*. This model was provided by deepset [deepset, 2021].

Starting from these three base models, we have initiated a series of model training tasks on the RCI Cluster, varying in the batch size and the overall number of epochs. The latter was typically not of an overwhelming significance, as, given the relatively small train split of the ČTK dataset, models soon started to overfit – see Figure 7.1. In such cases, we kept the dev-optimal model, tossing the later epochs.

epoch	train acc.	dev acc.
0.	0.80	0.77
1.	0.93	**0.87**
2.	0.96	0.81
3.	0.98	0.85
4.	0.98	0.84
⋮	⋮	⋮
100.	**1.00**	0.81

Table 7.1: The progress of XLM-RoBERTa@SQuAD2_bs4 in the train and dev accuracies during 100 epochs of training on the ČTK dataset

Despite that, we have set a strong baseline for the future NLI experiments on the ČTK datasets with our XLM-RoBERTa model scoring an F-value of **0.86**. For reference, the [Thorne et al., 2018a] baseline scored 80.82% *accuracy* in a similar setting (*NearestP* – nearest page for a NEI context) on Wikipedia, using a Decomposable Attention model.

ČTK v2 CSV split:		test		dev	
model	|*batch*|	**micro-F_1**	**macro-F_1**	**micro-F_1**	**macro-F_1**
SlavicBERT	2	*0.743*	0.700	0.771	0.735
SlavicBERT	5	*0.741*	0.702	0.782	0.757
mBERT	3	*0.727*	0.686	0.710	0.667
mBERT	10	*0.743*	0.717	0.742	0.721
XLM-RoBERTa @ SQuAD2	2	*0.807*	0.769	0.842	0.815
XLM-RoBERTa @ SQuAD2	4	***0.855***	**0.840**	**0.866**	**0.849**
XLM-RoBERTa @ SQuAD2	7	*0.835*	0.819	0.851	0.838

Table 7.2: F-score (**micro-F_1**) comparison of our BERT-like models for the *NLI* task on the **ČTK** data, experimenting with different training *batch sizes*. *Coursive* decisive, **bold** best.

[7] https://gitlab.fel.cvut.cz/factchecking/nli

Furthermore, we render the confusion matrices for two of our strongest models – the XLM-RoBERTa and the SlavicBERT, to see the distribution of their test-set *(mis-)classifications* – we see that the main disadvantage of SlavicBERT compared to the stronger XLM-RoBERTa is the understanding of the `REFUTES` annotations.

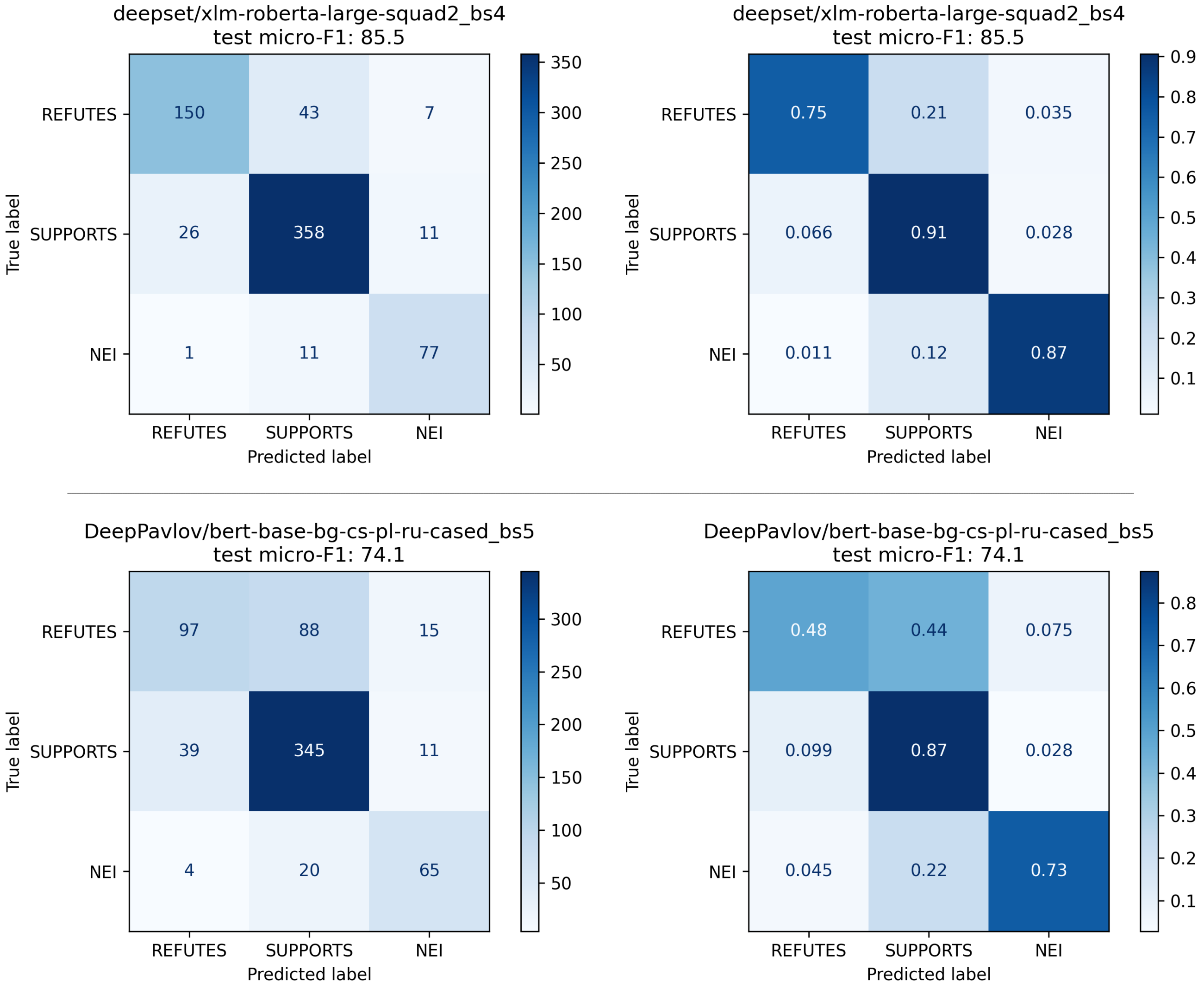


Figure 7.1: The confusion matrices of XLM-RoBERTa finetuned on *SQuAD2* and the SlavicBert models for Natural Language Inference on ČTK test data

7.4.3 Machine-translated NLI corpora

To address the *overfitting* (Table 7.1) issue in our future experiments, our team at AIC has internally obtained a machine-translated Czech localization for each of the corpora listed in 7.2, using the Google Translate API. The scheme is simpler than that from 3.2, as their formats are closely resemblant to the `nli` FCheck export (Figure 5.4), i.e., a set of plain text pairs and their labels.

In future, the Czech SNLI, MultiNLI, ANLI or FEVERNLI datasets could be either used directly to augment the ČTK NLI train dataset, or to construct a *down-stream task* for NLI in Czech.

In the weeks subsequent to the submission of this thesis, we will examine the licensing of the aforementioned corpora, and, if allowed, publish their Czech localizations in a cloud storage[8] to supplement this thesis.

7.5 Experiments Wrapup

In our last full chapter, we have used the data collected during our annotation experiments in Chapter 4 to train a round of Czech Natural Language Inference classifiers. The strongest of them, XLM-RoBERTa sets a vital benchmark for its future successors, and is ready to be experimentally used in the production environment, provided there is a reliable *Document Retriever* to feed its input (Figure 1.2).

We attach the experimental notebooks[9] used to train and validate our models, as well as the resulting model[10] in a hope for reproducibility of our experiments, however, the code quality is incomparable with the PHP application from Chapter 4.

[8] http://bertik.net/nli_corpora

[9] https://gitlab.fel.cvut.cz/factchecking/nli

[10] http://bertik.net/ctk-xlm-roberta

Chapter 8
Conclusion

Our work has addressed the lack of a fact-verification dataset in Czech in two ways:

Firstly, it established a scheme of transferring an English ENWiki-corpus-based FEVER dataset to the Czech language using Machine Translation and the cross-lingual mapping of WikiMedia API, obtaining a set of a total of 127K translated claims along with their veracity labels and evidence within the CSWiki corpus, which we call the FEVER CS dataset.

Secondly, we prepared a series of human-annotation experiments, that were conducted with 163 annotators, utilizing the collaboration with the Faculty of Social Sciences of Charles University towards collecting about 10K *Claim* and *Label* data points stemming from our application-specific ČTK Archive knowledge base, achieving an inter-annotator aggreement of 0.63, measured using the 4-way Fleiss' κ.

For the annotations, we have built a novel annotation platform from the ground up, naming it the FCheck Annotations Platform and publishing it as an open-source project. Subsequently, we have used our platform to export the novel ČTK dataset, that contains 3,295 textual claims along with their veracity labeling and ČTK-based sets of conclusive evidence, extracted from the results earlier annotation experiments.

Finally, we deem this dataset eligible for training statistical models for the task of Natural Language Inference, demonstrating the usage of our data on transfer-learning a triple of Transformer networks – XLM-RoBERTa, SlavicBERT and multilingualBERT, the first of which scores **85.5%** micro-F_1 on the ČTK claim veracity labeling task.

8.1 Proposed solutions

At the end of every chapter, we provide a textual *wrapup* of its result, along with remarks on its reproducibility, and, where possible, a ready-made solution in form of an open-source code, or prebuilt models and datasets shared through a public cloud-storage link.

This is to encourage any future research on the topic, as well as to challenge our results and their credibility.

8.2 Future research goals

Our work at AIC FactCheck is far from over. After the publication of the ČTK dataset and the baseline NLI classifier, we are about to pursue some of the following goals that arised from the findings in previous chapters:

1. The solution for the *overfitting* issue from Chapter 7 should be examined, using some of the attached localized SNLI, MultiNLI, ANLI and FEVER-NLI sets, as outlined in the previous Chapter *wrapup*

2. The novel monolingual Czech **CZERT** model is to be trained and examined on the same tasks as the other models from the Chapter 7

3. **The FEVER CS Baseline** end-to-end containerized pipeline should be updated with our resulting models and that of [Rýpar, 2021] for the production purposes

4. The set of T_{2b} **Claim Mutations** (Figure 4.1) collected by the FCheck platform is to be examined and challenged with the dataset balance in mind, as the same set of mutation tasks yielded a significantly label-unbalanced dataset to *us* (Chapter 5), to [Thorne et al., 2018a] and to [Binau and Schulte, 2020], all of them in favour of the `SUPPORTS annotation`

Bibliography

[Allcott and Gentzkow, 2017] Allcott, H. and Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36.

[Arkhipov et al., 2019] Arkhipov, M., Trofimova, M., Kuratov, Y., and Sorokin, A. (2019). Tuning multilingual transformers for language-specific named entity recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 89–93, Florence, Italy. Association for Computational Linguistics.

[Astrakhan et al., 2021] Astrakhan, Y., Kattouw, R., Vasiliev, V., Reed, B. T. M. S., and Jorsch, B. (2021). Api: Main page. `mediawiki.org/wiki/API:Main_page`. Accessed: 2021-05-11.

[Barua et al., 2020] Barua, Z., Barua, S., Aktar, S., Kabir, N., and Li, M. (2020). Effects of misinformation on covid-19 individual responses and recommendations for resilience of disastrous consequences of misinformation. *Progress in Disaster Science*, 8:100119.

[Binau and Schulte, 2020] Binau, J. and Schulte, H. (2020). Danish fact verification: An end-to-end machine learning system for automatic fact-checking of danish textual claims. `https://www.derczynski.com/itu/docs/fever-da_jubi_hens.pdf`.

[Bowman et al., 2015] Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. *CoRR*, abs/2005.14165.

[Buchanan and Benson, 2019] Buchanan, T. and Benson, V. (2019). Spreading disinformation on facebook: Do trust in message source, risk propensity, or personality affect the organic reach of "fake news"? *Social Media + Society*, 5(4):2056305119888654.

[Chang et al., 2020] Chang, W., Yu, F. X., Chang, Y., Yang, Y., and Kumar, S. (2020). Pre-training tasks for embedding-based large-scale retrieval. *CoRR*, abs/2002.03932.

[Chatzikyriakidis et al., 2017] Chatzikyriakidis, S., Cooper, R., Dobnik, S., and Larsson, S. (2017). An overview of natural language inference data collection: The way forward? In *Proceedings of the Computing Natural Language Inference Workshop*, page 2.

[Chen et al., 2017] Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading wikipedia to answer open-domain questions. *CoRR*, abs/1704.00051.

[Chen, 1976] Chen, P. P.-S. (1976). The entity-relationship model—toward a unified view of data. *ACM Trans. Database Syst.*, 1(1):9–36.

[Cheng et al., 2016] Cheng, J., Dong, L., and Lapata, M. (2016). Long short-term memory-networks for machine reading. *CoRR*, abs/1601.06733.

[Cohen, 1960] Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

[Conneau et al., 2019] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

[Drchal, 2020] Drchal, J. (2020). Anotace dat pro ověřování faktů nad databází článků Čtk. `https://fcheck.fel.cvut.cz/2020_fcheck_anotace.pdf`. [Online; accessed 13-May-2021].

[Drchal and Ullrich, 2020] Drchal, J. and Ullrich, H. (2020). CTU FEE GitLab – Fact Checking Experimental (Honza Drchal). `https://gitlab.fel.cvut.cz/factchecking/drchajan`. [Online; accessed 14-May-2021].

[Dědková, 2021] Dědková, B. (2021). Multi-stage methods for document retrieval in the czech language.

[Fleiss, 1971] Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

[Gažo, 2021] Gažo, A. (2021). Algorithms for document retrieval in czech language supporting long inputs.

[Hanselowski et al., 2018] Hanselowski, A., Zhang, H., Li, Z., Sorokin, D., Schiller, B., Schulz, C., and Gurevych, I. (2018). Ukp-athene: Multi-sentence textual entailment for claim verification. *arXiv preprint arXiv:1809.01479.*

[Jeffries et al., 2001] Jeffries, R., Anderson, A., and Hendrickson, C. (2001). *Extreme Programming Installed.* XP series. Addison-Wesley.

[Khattab and Zaharia, 2020] Khattab, O. and Zaharia, M. (2020). Colbert: Efficient and effective passage search via contextualized late interaction over BERT. *CoRR*, abs/2004.12832.

[Kiss and Strunk, 2006] Kiss, T. and Strunk, J. (2006). Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32:485–525.

[Košarko et al., 2019] Košarko, O., Variš, D., and Popel, M. (2019). LINDAT translation service. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

[Krippendorff, 2013] Krippendorff, K. (2013). *Content Analysis: An Introduction to Its Methodology.* SAGE Publications.

[Lan et al., 2019] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

[Landis and Koch, 1977] Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

[Lazer et al., 2018] Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D., Schudson, M., Sloman, S. A., Sunstein, C. R., Thorson, E. A., Watts, D. J., and Zittrain, J. L. (2018). The science of fake news. *Science*, 359(6380):1094–1096.

[Liu et al., 2019] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

[Malon, 2018] Malon, C. (2018). Team papelo: Transformer networks at fever. In *Proceedings of the EMNLP First Workshop on Fact Extraction and Verification.*

[Nie et al., 2019a] Nie, Y., Chen, H., and Bansal, M. (2019a). Combining fact extraction and verification with neural semantic matching networks. In *Association for the Advancement of Artificial Intelligence (AAAI).*

[Nie et al., 2019b] Nie, Y., Williams, A., Dinan, E., Bansal, M., Weston, J., and Kiela, D. (2019b). Adversarial NLI: A new benchmark for natural language understanding. *CoRR*, abs/1910.14599.

[Nuseibeh and Easterbrook, 2000] Nuseibeh, B. and Easterbrook, S. (2000). Requirements engineering: A roadmap. In *Proceedings of the Conference on The Future of Software Engineering*, ICSE '00, page 35–46, New York, NY, USA. Association for Computing Machinery.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

[Pitr, 2020] Pitr, M. (2020). CTU FEE GitLab – Experimental: Michal Pitr. `https://gitlab.fel.cvut.cz/factchecking/experimental-michal_pitr`. [Online; accessed 14-May-2021].

[Poliak, 2020] Poliak, A. (2020). A survey on recognizing textual entailment as an NLP evaluation. *CoRR*, abs/2010.03061.

[Popel et al., 2020] Popel, M., Tomkova, M., Tomek, J., Kaiser, Ł., Uszkoreit, J., Bojar, O., and Žabokrtský, Z. (2020). Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals. *Nature Communications*, 11(4381):1–15.

[Post, 2018] Post, M. (2018). A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

[Přibáň et al., 2019] Přibáň, P., Hercig, T., and Steinberger, J. (2019). Machine Learning Approach to Fact-checking in West Slavic Languages. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2019*, Varna, Bulgaria. INCOMA Ltd.

[Raffel et al., 2019] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

[Rajpurkar et al., 2016] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250.

[Reimers and Gurevych, 2019] Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084.

[Rýpar, 2021] Rýpar, M. (2021). Methods of document retrieval for fact-checking. `https://www.overleaf.com/read/thbvcjvvvfjp`. [Online; accessed 21-May-2021].

[Shu et al., 2018] Shu, K., Wang, S., Le, T., Lee, D., and Liu, H. (2018). Deep headline generation for clickbait detection.

[Sido et al., 2021] Sido, J., Pražák, O., Přibáň, P., Pašek, J., Seják, M., and Konopík, M. (2021). Czert – czech bert-like model for language representation.

[STEM, 2021] STEM (2021). Mýtům a konspiracím o covid-19 věří více než třetina české internetové populace | stem.cz. `https://www.stem.cz/mytum-a-konspiracim-o-covid-19-veri-vice-nez-tretina-ceske-internetove-populace/`. Accessed: 2021-05-03.

[Straková et al., 2019] Straková, J., Straka, M., and Hajic, J. (2019). Neural architectures for nested NER through linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331, Florence, Italy. Association for Computational Linguistics.

[DeepL, 2021] DeepL (2021). Deepl translator. `https://www.deepl.com/en/translator`. Accessed: 2021-05-09.

[DeepPavlov, 2021] DeepPavlov (2021). Deeppavlov: an open source conversational ai framework. `https://deeppavlov.ai/`. [Online; accessed 21-May-2021].

[deepset, 2021] deepset (2021). deepset - cutting-edge nlp solutions. `https://deepset.ai/`. [Online; accessed 21-May-2021].

[Google, 2021] Google (2021). Cloud translation | google cloud. `https://cloud.google.com/translate`. Accessed: 2021-05-09.

[Wikipedia, 2021a] Wikipedia (2021a). Wikipedia:Encyclopedic style — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Wikipedia:Encyclopedic_style&oldid=1009871271`. [Online; accessed 12-May-2021].

[Wikipedia, 2021b] Wikipedia (2021b). Wikipedia:Wikipedia is not a reliable source — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Wikipedia%3AWikipedia%20is%20not%20a%20reliable%20source&oldid=1017600260`. [Online; accessed 12-May-2021].

[drawSQL, 2021] drawSQL (2021). fcheck | drawsql. `https://drawsql.app/sir/diagrams/fcheck`. [Online; accessed 12-May-2021].

[The European Commission, 2015] The European Commission (2015). Ects users' guide 2015. `https://op.europa.eu/en/publication-detail/-/publication/da7467e6-8450-11e5-b8b7-01aa75ed71a1`. [Online; accessed 12-May-2021].

[Thorne and Vlachos, 2019] Thorne, J. and Vlachos, A. (2019). Adversarial attacks against Fact Extraction and VERification.

[Thorne et al., 2018a] Thorne, J., Vlachos, A., Christodoulopoulos, C., and Mittal, A. (2018a). FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

[Thorne et al., 2018b] Thorne, J., Vlachos, A., Cocarascu, O., Christodoulopoulos, C., and Mittal, A. (2018b). The Fact Extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*.

[Thorne et al., 2019] Thorne, J., Vlachos, A., Cocarascu, O., Christodoulopoulos, C., and Mittal, A. (2019). The FEVER2.0 shared task. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 1–6, Hong Kong, China. Association for Computational Linguistics.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

[Wang et al., 2020] Wang, B., Wang, S., Cheng, Y., Gan, Z., Jia, R., Li, B., and Liu, J. (2020). Infobert: Improving robustness of language models from an information theoretic perspective. *CoRR*, abs/2010.02329.

[Wang et al., 2021] Wang, S., Fang, H., Khabsa, M., Mao, H., and Ma, H. (2021). Entailment as few-shot learner.

[Whistler, 2020] Whistler, K. (2020). Unicode normalization forms. Unicode Standard Annex 15.

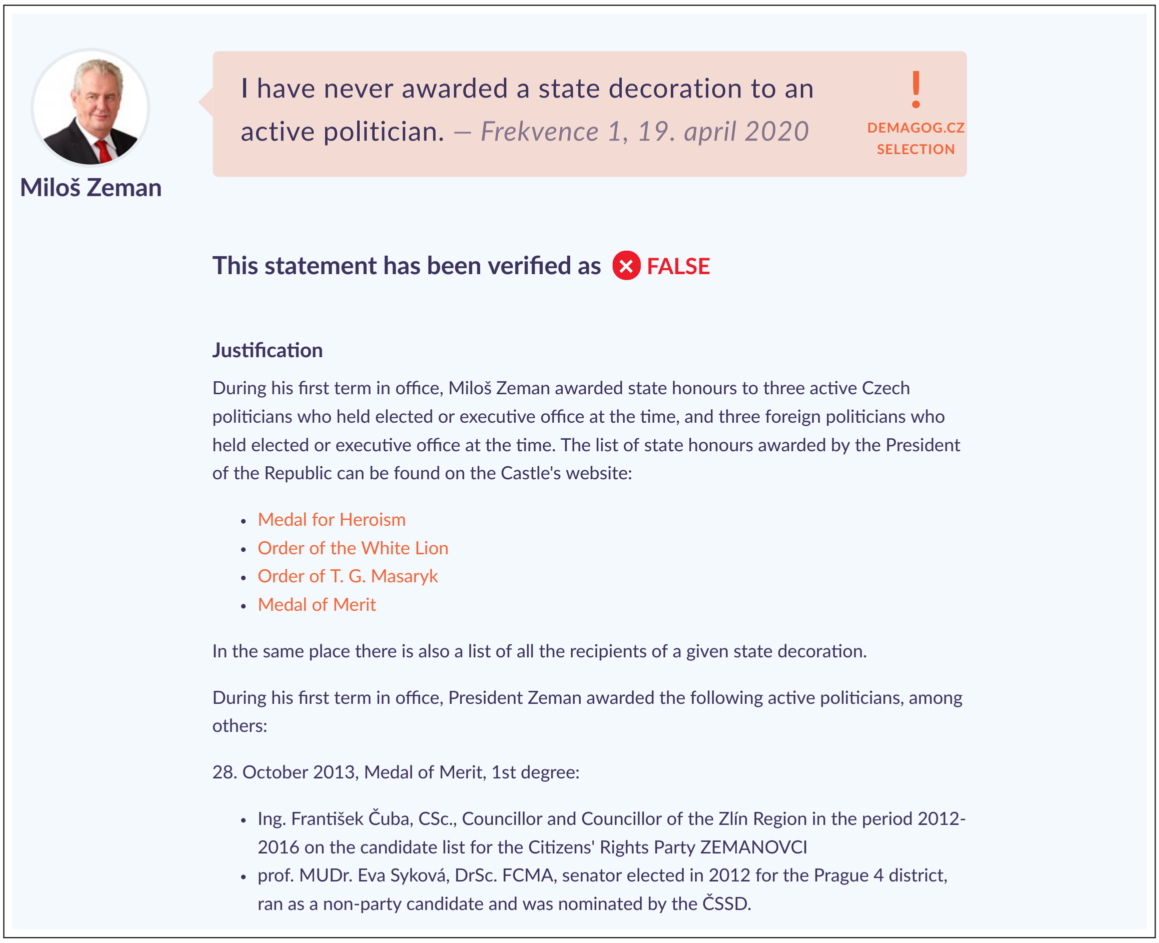
[Williams et al., 2018] Williams, A., Nangia, N., and Bowman, S. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

[Wolf et al., 2019] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J. (2019). Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Appendices

Appendix A
Czech-English data translations

A.1 Translated figures



Miloš Zeman

I have never awarded a state decoration to an active politician. — *Frekvence 1, 19. april 2020*

!
DEMAGOG.CZ
SELECTION

This statement has been verified as ✕ FALSE

Justification

During his first term in office, Miloš Zeman awarded state honours to three active Czech politicians who held elected or executive office at the time, and three foreign politicians who held elected or executive office at the time. The list of state honours awarded by the President of the Republic can be found on the Castle's website:

- Medal for Heroism
- Order of the White Lion
- Order of T. G. Masaryk
- Medal of Merit

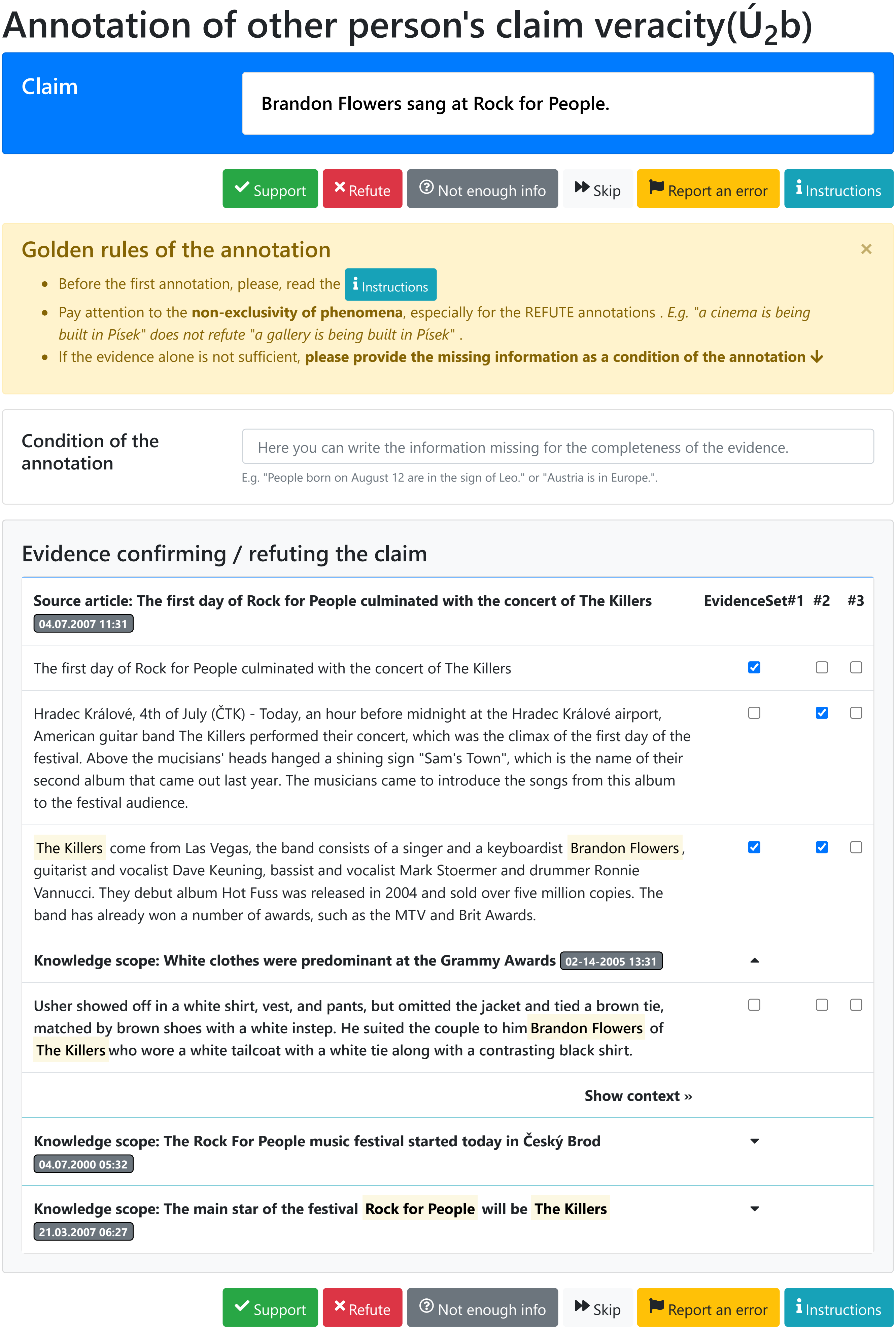
In the same place there is also a list of all the recipients of a given state decoration.

During his first term in office, President Zeman awarded the following active politicians, among others:

28. October 2013, Medal of Merit, 1st degree:

- Ing. František Čuba, CSc., Councillor and Councillor of the Zlín Region in the period 2012-2016 on the candidate list for the Citizens' Rights Party ZEMANOVCI
- prof. MUDr. Eva Syková, DrSc. FCMA, senator elected in 2012 for the Prague 4 district, ran as a non-party candidate and was nominated by the ČSSD.

Figure A.1: Translated fact verification from Czech portal Demagog.cz – original in Figure 1.1



Annotation of other person's claim veracity(Ú$_2$b)

Claim Brandon Flowers sang at Rock for People.

Golden rules of the annotation

- Before the first annotation, please, read the Instructions
- Pay attention to the **non-exclusivity of phenomena**, especially for the REFUTE annotations . *E.g. "a cinema is being built in Písek" does not refute "a gallery is being built in Písek"* .
- If the evidence alone is not sufficient, **please provide the missing information as a condition of the annotation ↓**

Condition of the annotation

Here you can write the information missing for the completeness of the evidence.

E.g. "People born on August 12 are in the sign of Leo." or "Austria is in Europe.".

Evidence confirming / refuting the claim

Source article: The first day of Rock for People culminated with the concert of The Killers 04.07.2007 11:31	**EvidenceSet#1**	**#2**	**#3**
The first day of Rock for People culminated with the concert of The Killers	☑	□	□
Hradec Králové, 4th of July (ČTK) - Today, an hour before midnight at the Hradec Králové airport, American guitar band The Killers performed their concert, which was the climax of the first day of the festival. Above the mucisians' heads hanged a shining sign "Sam's Town", which is the name of their second album that came out last year. The musicians came to introduce the songs from this album to the festival audience.	□	☑	□
The Killers come from Las Vegas, the band consists of a singer and a keyboardist Brandon Flowers, guitarist and vocalist Dave Keuning, bassist and vocalist Mark Stoermer and drummer Ronnie Vannucci. They debut album Hot Fuss was released in 2004 and sold over five million copies. The band has already won a number of awards, such as the MTV and Brit Awards.	☑	☑	□
Knowledge scope: White clothes were predominant at the Grammy Awards 02-14-2005 13:31	▲		
Usher showed off in a white shirt, vest, and pants, but omitted the jacket and tied a brown tie, matched by brown shoes with a white instep. He suited the couple to him Brandon Flowers of The Killers who wore a white tailcoat with a white tie along with a contrasting black shirt.	□	□	□
Show context »			
Knowledge scope: The Rock For People music festival started today in Český Brod 04.07.2000 05:32	▼		
Knowledge scope: The main star of the festival Rock for People will be The Killers 21.03.2007 06:27	▼		

Figure A.2: The labelling interface of FCheck platform. Czech original in Figure 4.5

Appendix B
Acronyms

BERT Bidirectional Encoder Representations from Transformers

FEVER Fact Extraction and Verification – series of Shared tasks focused on fact-checking

CLI Command-Line Interface

NLI Natural Language Inference

ČTK Czech Press Agency