WikiCheck: An End-to-end Open Source Automatic Fact-Checking API based on Wikipedia

Mykola Trokhymovych, Diego Saez Trumper
Ukrainian Catholic University, Wikimedia Foundation

Abstract

With the growth of fake news and disinformation, the NLP community has been working to assist humans in fact-checking. However, most academic research has focused on model accuracy without paying attention to resource efficiency, which is crucial in real-life scenarios. In this work, we review the State-of-the-Art datasets and solutions for Automatic Fact-checking and test their applicability in production environments. We discover overfitting issues in those models, and we propose a data filtering method that improves the model's performance and generalization. Then, we design an unsupervised fine-tuning of the Masked Language models to improve its accuracy working with Wikipedia. We also propose a novel query enhancing method to improve evidence discovery using the Wikipedia Search API. Finally, we present a new fact-checking system, the WikiCheck API that automatically performs a facts validation process based on the Wikipedia knowledge base. It is comparable to SOTA solutions in terms of accuracy and can be used on low-memory CPU instances.

Problem formulation

- **End-to-end fact-checking:**

Given the claim, classify it as true or false and provide evidence for your reasoning from a reliable knowledge base

- **Natural language inference (NLI):**

Given two texts (claim and hypothesis), decide if the hypothesis supports the initial claim, refutes it, or does not relate to it.

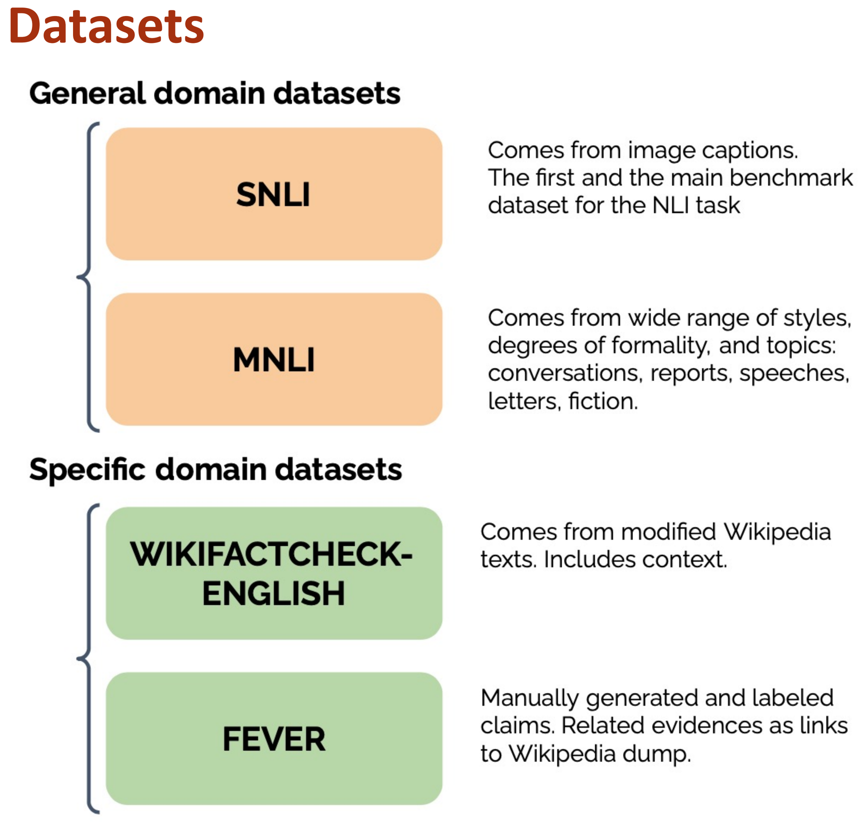
Open problems

- The efficiency of NLI models is not considered in previous research
- Lack of high-quality NLI datasets for model training
- Software architecture for end-to-end fact-checking

Research goals

- Analyse NLI datasets. Define the specific data features and limitation, design a methodology for data quality improvement.
- Experiment with NER models usage for information retrieval stage
- Build accurate and efficient domain specific sentence-based NLI model. Experiment with unsupervised learning and transfer learning.
- **Implement an open-source end-to-end fact-checking API.**

Datasets



SNLI & MNLI. Data sample

Dataset	Claim	Hypothesis	Label
MNLI	The Old One always comforted Ca'daan, except today.	Ca'daan knew the Old One very well.	neutral
MNLI	At the other end of Pennsylvania Avenue, people began to line up for a White House tour.	People formed a line at the end of Pennsylvania Avenue.	entailment
SNLI	A man inspects the uniform of a figure in some East Asian country.	The man is sleeping	contradiction
SNLI	An older and younger man smiling.	Two men are smiling and laughing at the cats playing on the floor.	neutral

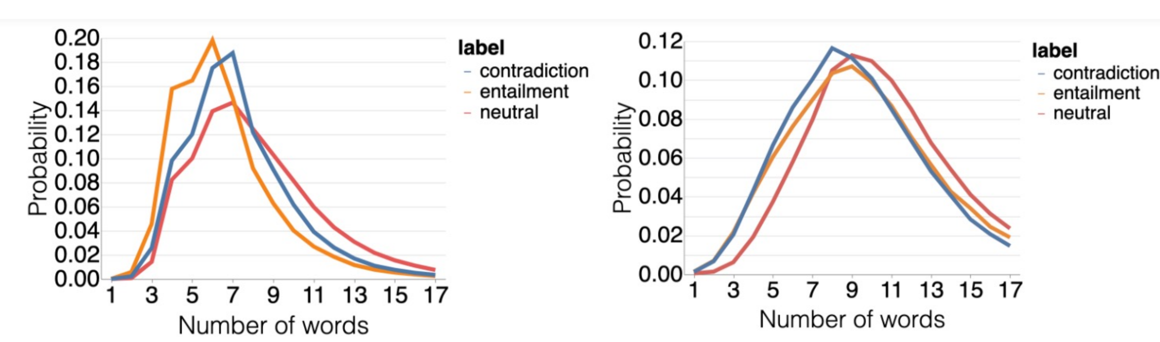
FEVER. Data sample

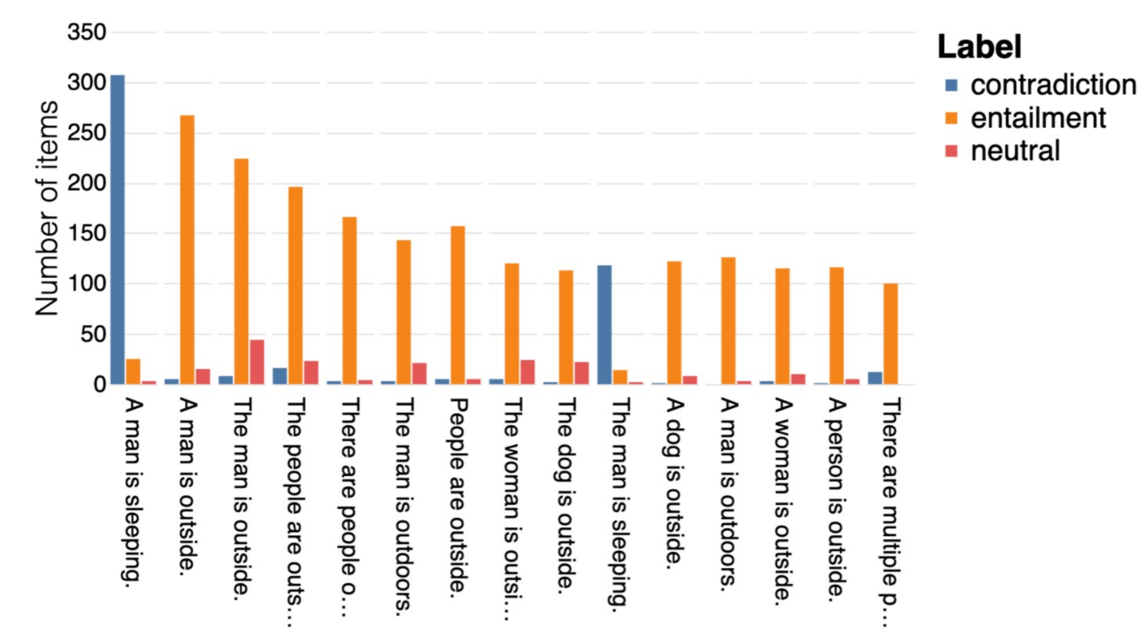

```
{"id": 75397,
 "verifiable": "VERIFIABLE",
 "label": "SUPPORTS",
 "claim": "Nikolaj Coster-Waldau worked with the Fox Broadcasting Company.",
 "evidence": [[[92206, 104971, "Nikolaj_Coster-Waldau", 7],
              [92206, 104971, "Fox_Broadcasting_Company", 0]]]}
```


+ Wikipedia dump (2017)

Annotation artifacts

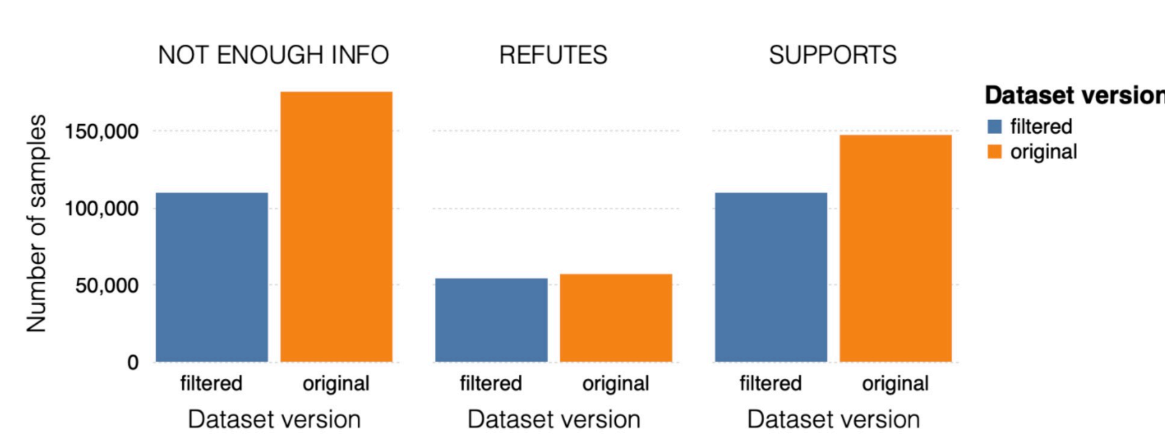
Distributions of length of hypothesis in training dataset for sample of different label is different. We observe disbalance across labels of samples with the same hypothesis for SNLI dataset. The same pattern exists for FEVER.





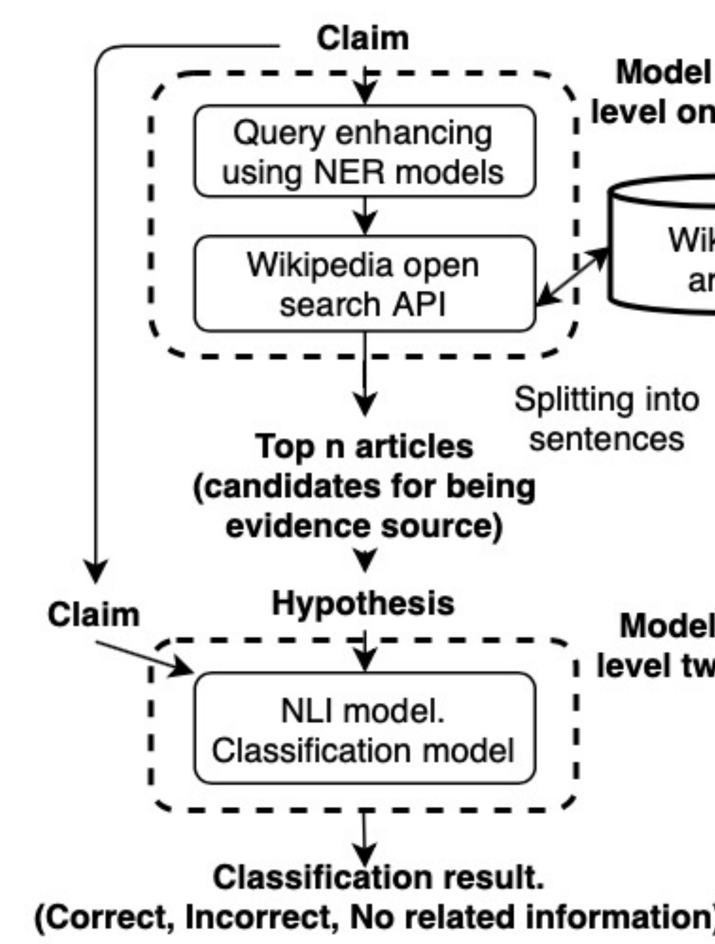
Data preparation. Filtering

- Filtered out absolute duplicates by fields 'claim' and 'hypothesis'. (8.8% reduced)
- Balancing distribution of SUPPORTS/REFUTES classes among hypothesis sentences. (6.9% reduced)
- Undersample NOT ENOUGH INFO class samples to the amount of major class. (12.2% reduced)

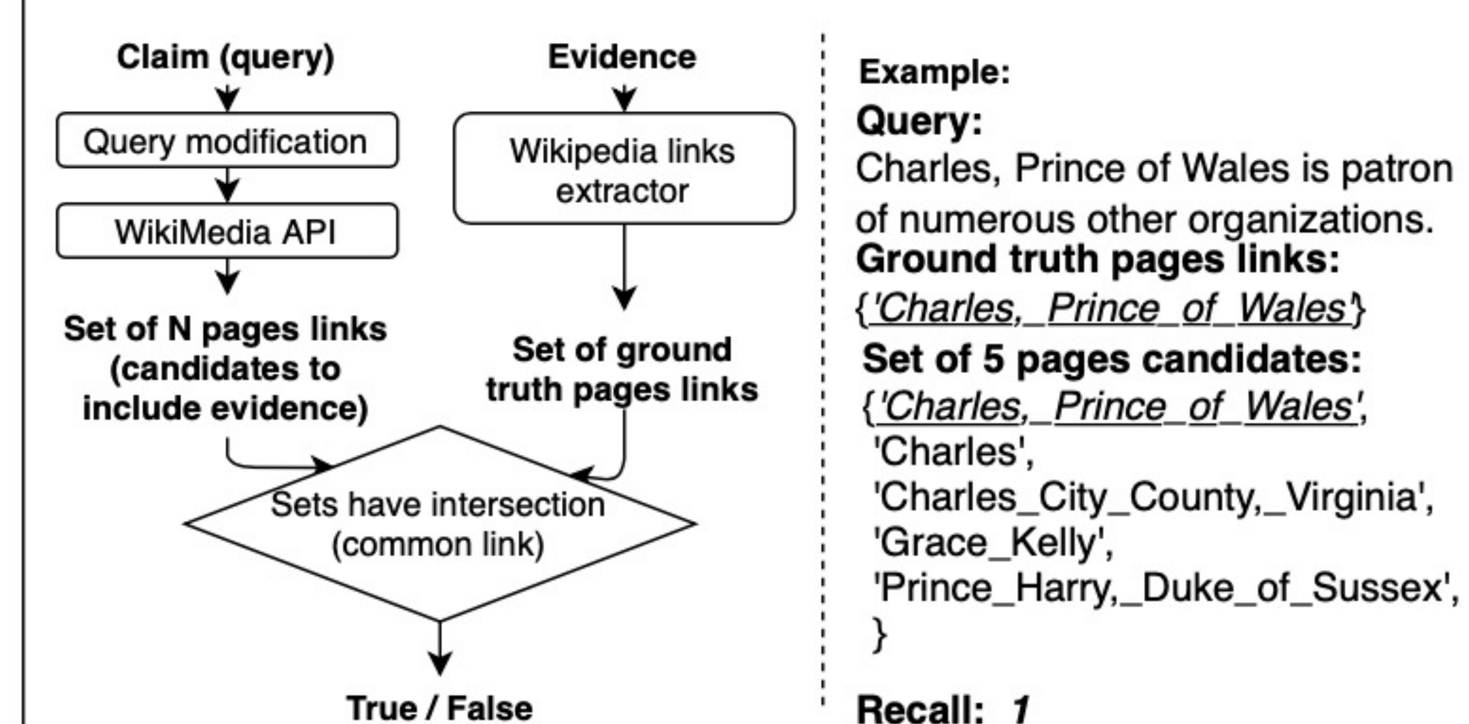


WikiCheck system architecture

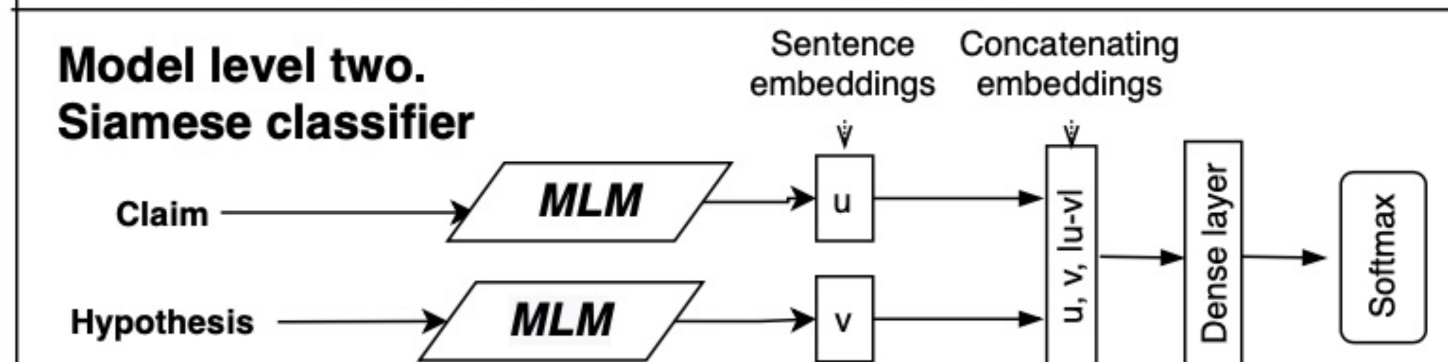
General architecture



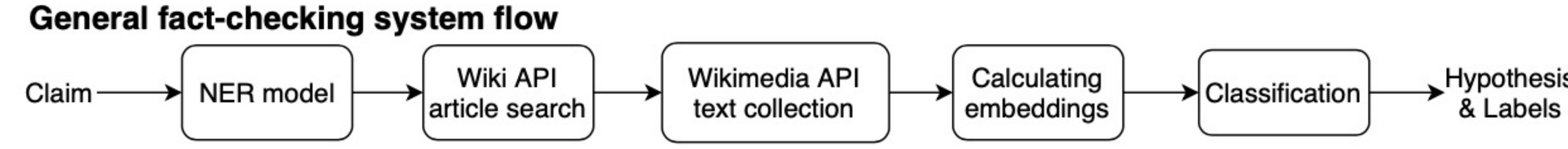
Model level one validation



Model level two. Siamese classifier



General fact-checking system flow



Experiments & Results

- **Improving the performance of search**

As a result we decided to use the "Flair ner-fast NER separate N=3" configuration. It provides high accuracy of 0.879 AR with only 6.27 candidates returned.

- **Model Generalization**

We trained a NLI model on the MNLI, tested on SNLI and MNLI testing set. We found that the accuracy decays between 11% to 16% depending on the MLM used.

Also, we train models on SNLI and then fine-tune the last layer on MNLI or FEVER train set and test on the corresponding test set. We also trained individual models for each dataset with all layers unfrozen and compared performance with the transfer learning approach.

Model	**MNLI adapted vs. full train**	**FEVER adapted vs. full train**
bert-base-uncased	64.8% / 76.1%	70.1% / 79.81%
bart-base	67.6% / 77.85%	74.4% / 85.24%
bert-base-uncased + fine tuned	65.4% / 76.29%	69.7% / 82.45%
bart-base + fine tuned	68.1% / 77.35%	73% / 85.62%

- **Training model on filtered FEVER vs. original (cleaned) dataset**

Model	**original vs. filtered**	**original vs. filtered R&S only**
albert-base	71.85% / 72.40%	67.11% / 68.46%
bert-base-uncased	71.67% / 73.04%	67.17% / 70.49%
bart-base	74.72% / 75.53%	68.11% / 71.47%
bert-base-uncased + fine tuned	71.76% / 73.38%	67.01% / 70.44%
bart-base + fine tuned	74.82% / **75.91%**	68.34% / 71.91%

Complete system accuracy & efficiency

Team/Name	FEVER rank	Evidence F1	FEVER score	Accuracy
UNC-NLP	1	0.5322	0.6398	0.6798
UCL MRG	2	0.3521	0.6234	0.6744
Athene	3	0.3733	0.6132	0.6522
Ohio St. Uni	7	0.5854	0.4322	0.4989
WikiCheck	-	0.3587	0.4307	0.5753
GESIS Cologne	8	0.1981	0.4058	0.5395

system parts	albert	bart	bert
NER_model	0.06 ± 0.02	0.06 ± 0.02	0.06 ± 0.02
wiki_search	0.39 ± 0.20	0.40 ± 0.20	0.39 ± 0.21
wiki_texts	2.40 ± 1.14	2.37 ± 1.06	2.30 ± 1.10
embedding_claim	0.07 ± 0.01	0.08 ± 0.02	0.07 ± 0.01
embedding_hypothesis	3.05 ± 1.36	3.18 ± 1.59	2.56 ± 1.28
classification	0.01 ± 0.01	0.01 ± 0.01	0.01 ± 0.01
total_time	5.97 ± 2.35	6.11 ± 2.47	5.41 ± 2.24

Contacts and links

Contact:

WikiCheck Github:

WikiCheck API: nli.wmcloud.org