Claim Generation for Fact Verification Models

Information Retrieval

Antwerp, Belgium 15th December 2023

Konstantina Ellina
Nada Chehbouni
Pablo de Vicente

Zero-shot Fact Verification

Zero-shot Fact Verification

Fact verification models are widely utilized across various domains, including journalism and scientific documentation. With the vast volume of information available today, combating misinformation is a significant challenge. Fact Verification (FV) algorithms serve as a tool to manage this information overload, helping to discern truth from falsehood by checking statements.

The challenge with Fact Verification (FV) models is the significant amount of data required for training, which must be specifically annotated for each field. This process demands substantial resources in terms of both finances and human labor. However, the reality is that obtaining extensive training data for every new domain necessitating fact verification is impractical. Generating this data often involves soliciting humans to formulate claims and seek supporting or opposing evidence—a task that proves exceedingly expensive.

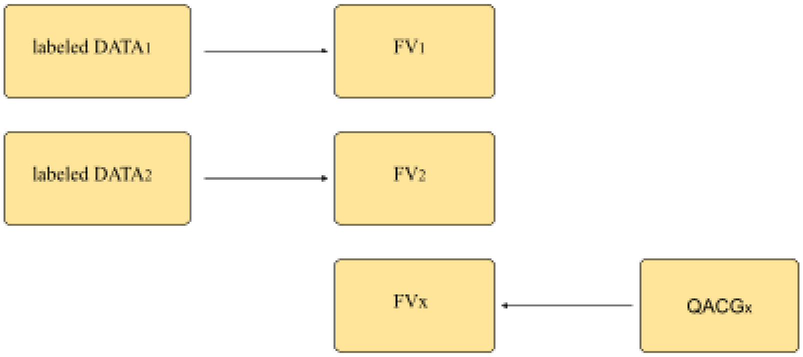


Figure 1: Manually labeling data is an expensive task needed to train specific-domain models. To solve this we use automatically generated claims

The paper addresses this problem by exploring the possibility of automatically generating large-scale (evidence, claim) pairs to train the fact verification model. It proposes a simple yet general framework Question Answering for Claim Generation (QACG) to generate three types of claims from any given evidence: 1) claims that are supported by the evidence, 2) claims that are refuted by the evidence, and 3) claims that the evidence does Not have Enough Information (NEI) to verify. We'll explore the specifics of how this framework achieves said goals in the following sections.

Zero shot implies that there is no human-annotated training example, the only data used for training is the one generated by the model. Using this technique the model achieves over 70% of the *F1* performance of a fully-supervised setting. By fine tuning the model with only 100 labeled examples, we further close the performance gap, achieving 89.1% of fully supervised performance. The above results show that pretraining the fact verification model with generated claims greatly reduces the demand for in-domain human annotation. When evaluating the model on an unbiased test set for FEVER, we find that training with generated claims also produces a more robust fact verification model.

Introduction of the FEVER Dataset

As the FEVER dataset is the one we will be using, it is only reasonable that we would give context on how it is constructed.

Creation of the FEVER Dataset

FEVER: Fact Extraction and VERification is presented as a large-scale dataset for claim verification. It consists of sentences that may be supported or refuted, along with a set of documents that can be used to verify them. The dataset contains a diverse set of claims and covers a wide range of topics, this is done with the intention of being a versatile dataset, and has its use on an extensive number of ML applications.

The dataset was constructed in two stages:

- **Claim Generation**
  For this specific task, a Wikipedia dump (as of June 2017) was processed and used to sample sentences from the introductory sections of ~50,000 pages. Annotators were then given these sentences and asked to generate a set of claims containing a single piece of information related to the original topic.

- **Claim Labeling**
  After claims were generated, the annotators labeled them as *SUPPORTED*, *REFUTED* or *NOTENOUGHINFO*. For the first two classes, evidence that either supports or refutes the claim had to be provided as well. In the case in which no amount of information from Wikipedia could support or refute the claim, the label *NOTENOUGHINFO* was used.

Baseline System Description

The entire system's workflow can be condensed into three main steps

- **Document Retrieval**. Done through a document retrieval component which returns the *k* most relevant documents for a query using cosine similarity between binned unigram and bigram Term Frequency Inverse Document Frequency (TF-IDF) vectors.

- **Sentence Selection.** This is achieved by ranking sentences by TF-IDF similarity to the claim and choosing the highest ranked ones. Note that this has to be validated using a metric of accuracy with respect to the development set. We further evaluate impact of sentence selection on the **RTE (Recognizing Textual Entailment)** module by predicting entailment given the original documents without sentence selection.

- **Recognizing Textual Entailment** in which two models for recognizing textual entailment were compared. The first one being a multi-layer perceptron (MLP) with a single hidden layer which uses term frequencies and TF-IDF cosine similarity between the claim and evidence as features, and the second one being a decomposable attention (DA) model between the claim and the evidence passage. The dataset could not be used for training purposes in this case since the NEI class had no evidence annotated to it. For this specific issue to be overcomed, training instances for this specific class were simulated through two methods: sampling a sentence from the nearest page (NEARESTP) to the claim as evidence using the document retrieval component, and sampling a sentence at random from Wikipedia (RANDOMS).

Evaluation

For each of the components mentioned above, tests were conducted. The results evidenced that the most challenging part to get right was to select suitable sentences as evidence.

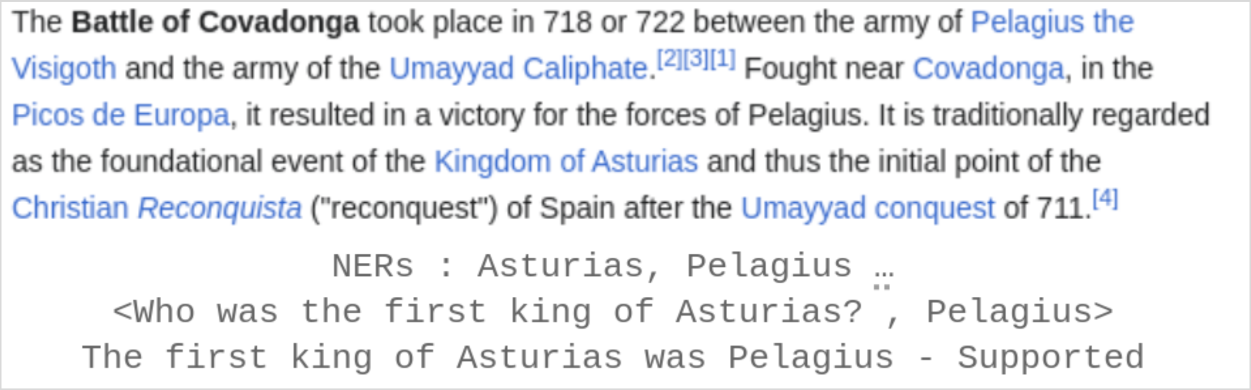
To evaluate the model, the test finds 5 documents closest to the claim along with its corresponding 5 concatenated sentences. Using the Decomposable Attention (DA) model, entailment is predicted for this sample data. If we adhere to the most strict evaluation metrics (in which the evidence provided had to be completely correct), the success rate was 31.87%. This really highlights how tricky it is to create this particular section of the model.

Claim Generation

Overview

In this section we will describe how the model works and what components it is made of. First we'll begin by defining the terms used

1. P : Original passage from which to extract information
2. (Q,A) : Questions and Answers generated from the passage
3. C : Claims are produced for each question according to the following rules:
   If P answers Q and A is correct then C = supported
   If P answers Q and A is incorrect then C = refuted
   If P can not answer Q then C = NEI (not enough information)
4. G : pretrained question generation model
5. M : pretrained QA to claim model
6. E : All entities contained in a passage



The **Battle of Covadonga** took place in 718 or 722 between the army of Pelagius the Visigoth and the army of the Umayyad Caliphate.[2][3][1] Fought near Covadonga, in the Picos de Europa, it resulted in a victory for the forces of Pelagius. It is traditionally regarded as the foundational event of the Kingdom of Asturias and thus the initial point of the Christian *Reconquista* ("reconquest") of Spain after the Umayyad conquest of 711.[4]

```
NERs : Asturias, Pelagius …
<Who was the first king of Asturias? , Pelagius>
The first king of Asturias was Pelagius - Supported
```

Figure 2: Example of passage,entity extraction, <question,answer> pair and supported claim

There are three major steps in the process as illustrated in Figure 2, the first of which is the Named Entity Recognition (**NER**).

The goal here is to extract Named Entities from the evidence context of each sample in the FEVER dataset. Named entities are essential for generating meaningful claims, they are often central to the meaning of a text, their extraction helps in understanding the key entities involved in a claim or statement. Named entity recognition involves identifying and classifying entities, such as persons, organizations, locations, and more, within a given text. This is used in later stages of the claim generation process. Here is an example of this procedure.

Context

"Over 120 colleges and universities are located in New York City , including Columbia University , New York University , and Rockefeller University , which have been ranked among the top 35 in the world."

Named entities extracted with their entity types

- "120" (CARDINAL)
- "New York City" (GPE)
- "Columbia University" (ORG)
- "New York University" (ORG)
- "Rockefeller University" (ORG)
- "35" (CARDINAL)

The entity types provide semantic labels of the extracted entities. In this example, "Cardinal" is some numerical values, "GPE" is geopolitical entities, such as countries, cities or states, and "ORG" is organizations.

When generating claims or statements from a given context, the entity types can be used to ensure that the generated claims are coherent and semantically meaningful. For example, in fact verification tasks, knowing the entity types can aid in determining the relevance of a claim to the provided evidence. If the claim involves a person receiving an award, and the entity types indicate that the mentioned award is a "WORK_OF_ART", we understand that the context of the text is related to creative works.

The extraction is executed with Stanza. Stanza is a Python natural language processing library that includes a Named Entity Recognizer (NER) among its tools. Stanza's NER component utilizes deep learning techniques to accurately identify and categorize named entities in multiple languages. In *Extract_NERs.py*, the processing of the text and the identification of the named entities are completed by the Stanza NLP library with the use of a Pipeline. The script iterates through the samples in the FEVER dataset, processes the context of each sample using Stanza, and extracts the identified named entities.

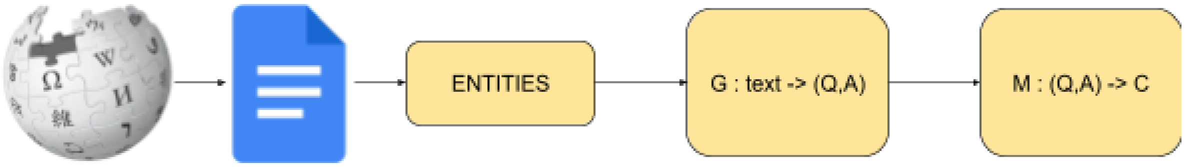


Figure 3: Sequence overview of claim generation

Then comes the **Question Generation** step**,** where, using a pre-trained Question Generation (QG) model, we generate (question, answer) pairs from the evidence. To accomplish this task, the BART model was enhanced through fine-tuning on the SQuAD (Stanford Question Answering Dataset) dataset and used. This model is designed to generate questions given input text and answer entities,

making it suitable for converting evidence into interrogative queries. That means that given an input text D and an answer A, the question generator outputs a question Q.
After the extraction of the named entities, the QG model is used to generate questions based on the provided sources (context) and answers (named entities). The results include pairs for each named entity, providing a set of questions that could be relevant to the given evidence.

Claims are then generated based on pairs of **Q**uestion and **A**nswers (**QA-to-Claim**). The QA-to-Claim Model takes the generated question Q and the corresponding answer A as inputs and produces a declarative sentence C for the (Q, A) pair. This model uses the BART paradigm and the SQuAD dataset to approach the problem as a sequence-to-sequence (seq2seq) task. The model may produce logical and contextually appropriate declarative sentences by taking into account the contextual relationships between the inquiry and answer, thanks to the seq2seq technique. Seq2seq models, like the ones used in our QA-to-Claim Model and Question Generator, are efficient at converting questions and other variable-length input sequences into meaningful and coherent output sequences, which makes them ideal for complex language creation tasks in natural language processing.

Claim generation is a crucial step in simulating diverse perspectives and queries relevant to a given piece of evidence. The generated claims encompass various perspectives—supported by evidence, refuted with alternatives, or acknowledging insufficient information—mimicking diverse user queries and enhancing the depth and breadth of information retrieval simulations. This intricate process underscores the versatility and adaptability of the claim generation mechanism in accommodating different types of queries associated with a given evidence set. The Claim Generation process involves utilizing pretrained models, specifically Sense2Vec and QA2D, to generate three types of claims – SUPPORTED, REFUTED, and NEI (Not Enough Info) – from the FEVER dataset.

- **Sense2Vec** model is used for finding answer replacements specifically for generating REFUTED claims. This model is pre-trained on a large corpus and has the capability to provide alternative representations for words or phrases. In this context, it assists in replacing certain answers in the generated (question, answer) pairs, introducing diversity, creativity and variability in the dataset for training, and evaluating the fact verification model.
- The pretrained **QA2D** model is used for generating claims based on the (question, answer) pairs obtained from the question generation step. QA2D is specifically designed for question-answering tasks. In this context, it aids in formulating claims by incorporating generated questions and answers into the desired claim format.

Supported claims

Given an evidence P, we use named entity recognition to identify all entities within P, denoted as E. For each entity $a \in E$, we treat each a in turn as an answer and generate a question $q = G(P, a)$ with the question generator. The question–answer pairs *(q, a)* are then sent to the QA-to-Claim model (QA2D) which then transforms existing question/answer pairs into claims.

Refuted claims

In order to generate refuted claims the procedure is as follows:

- **Answer Replacement:** The method starts with a question-answer pair (q, a). It uses "answer replacement" to replace the answer "a" with another entity "a_0" of the same type. The goal is to replace "a" with an incorrect answer to the question "q".
- **Selecting a Replacing Answer:** The paper uses the pre-trained *Sense2Vec* model to find the top-5 most similar phrases to the answer "a". Then, it randomly selects one of these phrases as the replacement answer "a_0".
- **Generating the Refuted Claim:** The new question-answer pair (q, a_0) is fed into a model called "QA-to-Claim". This model uses the information in the pair to generate a new statement, which is the refuted claim.
- **Avoiding Correct Replacements:** The method tries to avoid cases where "a_0" is actually a correct answer to the question "q". It defines rules to ensure that "a_0" has less overlap in words with the original answer "a". Even with these rules in place, it is not assured that correct replacements will be made, but they are few and far between so they are left as natural noise for the model.

NEI claims

- For **NEI** claims, additional context is extracted from Wikipedia using Stanza NLP. QA pairs are generated based on this extended context using the QG model, and QA2D is applied to produce claims.

For the **NEI**(**N**ot **E**nough **I**nfo) claims a question *q'* which is relevant but cannot be answered by *P* is generated. To this end, *P* is linked back to its original Wikipedia article *W* and the evidence is expanded with additional contexts *P+* extracted from Wikipedia using Stanza NLP. P+ consists of five randomly-retrieved sentences from *W* that are not present in *P*. Both *P* and *P+* are then concatenated and considered as the expanded evidence, based on which supported claims are generated given an entity in *P+* as the answer. This results in a claim relevant to but unverifiable by the original evidence *P*.

The **examples** below illustrate the sequence of the models used in the information retrieval pipeline, so we can completely understand the procedure followed in this report.

Named Entity Recognition (NER)

- Original Text: "Albert Einstein, a renowned physicist, was born on March 14, 1879, in Ulm, Germany."
- NER Output: [(Albert Einstein, PERSON), (March 14, DATE), (1879, DATE), (Ulm, GPE), (Germany, GPE)]

Question Generation (QG)

- Input to QG: The output of NER for a specific piece of evidence.
- QG Output: Generated (question, answer) pairs related to the identified entities.
    - "Who is Albert Einstein?" → "Albert Einstein is a PERSON."

    - “When was March 14 born?" → "March 14 is a DATE."

Claim Generation

- Input to Claim Generation: Original evidence (the text) along with the (question, answer) pairs generated by QG.
- Claim Generation Output: Synthetically generated claims categorized as SUPPORTED, REFUTED, or NEI.
    - **SUPPORTED** Claim Example: "Albert Einstein is a renowned physicist."
    - **REFUTED** Claim Example: "Stuttgart is the birthplace of Albert Einstein."
    - **NEI** Claim Example: "In addition to Germany, Albert Einstein had connections to other countries."

Evaluation

The evaluation of the fact verification depends on three different test sets based on FEVER. The idea is to contrast our generated claims with previously labeled information.

➔ **FEVER-S/R**: Since only the supported and refuted claims are labeled with gold evidence in FEVER, the claim–evidence pairs of these two classes from the FEVER test set are used for evaluation.

➔ **FEVER-Symmetric**: This is a carefully designed unbiased test set to detect the robustness of the fact verification model with the supported and refuted claims.

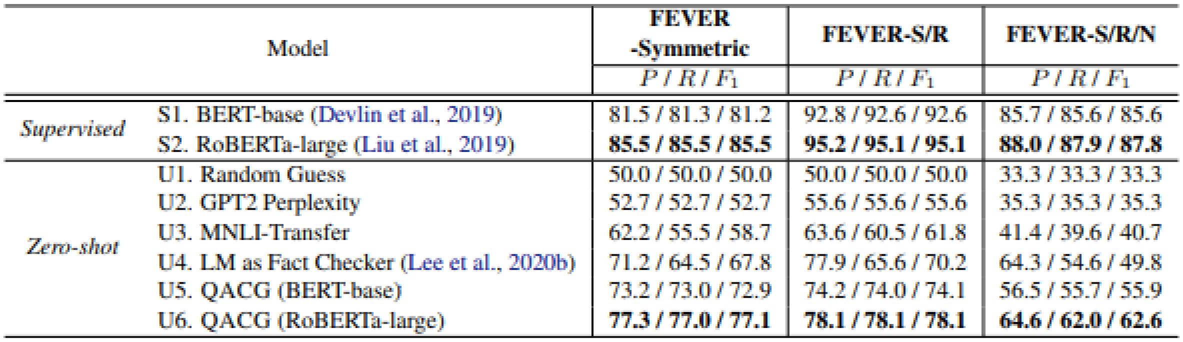
➔ **FEVER-S/R/N**: The full FEVER test set is used for a three-class verification. It is used to retrieve evidence sentences for NEI claims.

A BERT model and a RoBERTa-large model fine-tuned on the FEVER training set are the supervised models. The models are trained on the generated QACG-Filtered dataset (Question Answering for Claim Generation). For binary classification (FEVER-S/R and FEVER-Symmetric), only the supported and refuted claims are used for training, while for FEVER-S/R/N, the full training set is used.

Claims-evidence couples are difficult to interpret, but the BERT model—which has bidirectional context understanding—and the RoBERTa-large model—which improves and expands on BERT's architecture—are skilled at doing so. They efficiently sort through big databases and extract relevant proof by relying on prior knowledge. We use these models' ability to improve Precision, Recall, and F1 Score by training them on the FEVER training set along with the QACG-Filtered dataset.

Results

The best evaluation results are achieved with the RoBERTa-large model. Across all datasets and among some other zero-shot baselines, this model has the best results in Precision, Recall, and F1 Score. In Figure 4, we can see the baseline models and the supervised models with their evaluation results.



Model		FEVER -Symmetric	FEVER-S/R	FEVER-S/R/N
		P / R / F_1	P / R / F_1	P / R / F_1
Supervised	S1. BERT-base (Devlin et al., 2019)	81.5 / 81.3 / 81.2	92.8 / 92.6 / 92.6	85.7 / 85.6 / 85.6
	S2. RoBERTa-large (Liu et al., 2019)	**85.5 / 85.5 / 85.5**	**95.2 / 95.1 / 95.1**	**88.0 / 87.9 / 87.8**
Zero-shot	U1. Random Guess	50.0 / 50.0 / 50.0	50.0 / 50.0 / 50.0	33.3 / 33.3 / 33.3
	U2. GPT2 Perplexity	52.7 / 52.7 / 52.7	55.6 / 55.6 / 55.6	35.3 / 35.3 / 35.3
	U3. MNLI-Transfer	62.2 / 55.5 / 58.7	63.6 / 60.5 / 61.8	41.4 / 39.6 / 40.7
	U4. LM as Fact Checker (Lee et al., 2020b)	71.2 / 64.5 / 67.8	77.9 / 65.6 / 70.2	64.3 / 54.6 / 49.8
	U5. QACG (BERT-base)	73.2 / 73.0 / 72.9	74.2 / 74.0 / 74.1	56.5 / 55.7 / 55.9
	U6. QACG (RoBERTa-large)	**77.3 / 77.0 / 77.1**	**78.1 / 78.1 / 78.1**	**64.6 / 62.0 / 62.6**

Figure 4: Fact verification performance for supervised and zero-shot models on the three test sets

Transforming Question Answering Datasets Into Natural Language Inference Datasets (Creation of the QA2D Dataset)

The **QA-to-Claim Model** used in the '**Zero-shot Fact Verification by Claim Verification'** is a BART model that is finetuned on the **QA2D** dataset which is the subject of this paper (where QA2D means to combine **Q**uestions and **A**nswers into **D**eclarative answer sentences)

The database has been created based on the already existing SQuAD dataset which contains pairs of questions and answers based on Wikipedia articles. For each Q&A pair, a declarative sentence is generated, and based on entailment, an NLI example is then derived from the related Q&A example. Given the text passage P:

- If A is a correct answer to Q, then (P,D) is an entailed NLI pair.
- If A is an incorrect answer or Q cannot be answered using the information provided in P, then the generated sentence D is considered as not being implied by P, thus leading to (P,D) being a negative NLI pair.

In order to derive declarative sentences from question-answer pairs in this work, three different ways have been explored:

- **A rule-based system:** Relying on part-of-speech tagging (POS) and parsing accuracy, its overview can be seen in Figure 5. Generating grammatically correct sentences can still become a challenge when the answers in the <q,a> pair are named entities (*Where does Sam work?/WHO - Referring to the World Health Organization).* This will lead to sentences being generated without the proper syntax (*Sam works at WHO*)

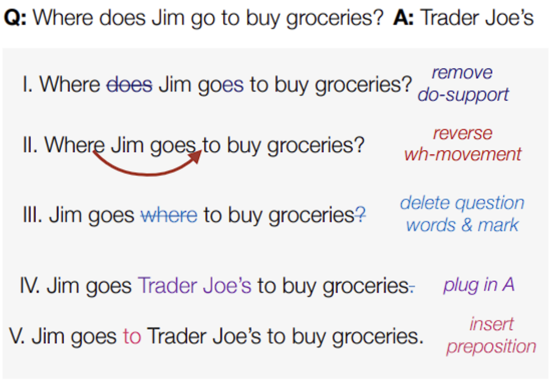

Q: Where does Jim go to buy groceries? A: Trader Joe's
I. Where ~~does~~ Jim goes to buy groceries?
remove do-support
II. Where Jim goes to buy groceries?
reverse wh-movement
III. Jim goes ~~where~~ to buy groceries~~?~~
delete question words & mark
IV. Jim goes Trader Joe's to buy groceries.
plug in A
V. Jim goes to Trader Joe's to buy groceries.
insert preposition

Figure 5: Syntactic transformations needed to perform QA2D using the rule-based system

- Crowdsourcing which would provide a collection of declarative sentences generated by humans. Participants were provided with question-answer pairs and asked to generate sentences from scratch in setup S, or an output generated by the rule-based model for them to edit in setup E. This step is required for the neural model that the authors of the paper decided to build for comparison purposes.

- ➔ A neural sequence model, with the aim of overcoming the rule-based system weaknesses.

Both the rule-based system and the neural sequence model performed relatively well but the latter had a slightly stronger overall performance.

Team Papelo: Transformer Networks at FEVER

Better system for FEVER challenge

In our case, the system that was introduced by the Papelo team was used as a retrieval method to collect evidence sentences for the NEI class.

The paper addresses the challenges posed by the FEVER challenge by developing a system centered around a transformer network-based entailment classifier. The primary goal is to enhance precision in classifying a wide range of potential evidence, allowing for improved recall in claim verification. The entailment module evaluates evidence statements individually, considering not only articles with the highest TF IDF scores but also additional ones based on named entities and capitalized expressions in the claim. This approach aims to overcome the complexities of FEVER's longer and more abstract sentences, increased prevalence of named entities, and the intricate retrieval process. The system strives to refine baseline approaches to retrieval and entailment by training a sharp entailment classifier. Comparative analysis reveals that the transformer network, with pre-trained weights, outperforms other models, especially in handling out-of-vocabulary words.

Entailment Module

The system's core is an entailment module built on a transformer network, a smart tool for understanding sequences. This transformer uses a separator to split the main idea from the argument and has twelve blocks to compare and understand information. It's like a smart reader trained on lots of languages from books. They fine-tuned this smart reader on FEVER examples to help it work better with tricky sentences, which is really good at figuring out complex connections in sentences. The transformer's adaptability in capturing intricate dependencies within sequential data proves crucial in the specific context of entailment classification, particularly when addressing the distinctive challenges presented by the FEVER dataset.

Improvements Over Baseline

There are numerous improvements that the Papelo team accomplishes over the base FEVER dataset. We'll highlight the most impactful causes, but we strongly suggest reading the entire paper.

➔ The system improves upon the baseline FEVER approach, which concatenated five premise statements and assessed them together. Instead, this system evaluates each premise statement individually, allowing for a more precise understanding.

➔ Training data is collected by selecting the top five sentences with the highest relevance (TF IDF score) against a claim from Wikipedia pages. This creates an entailment problem, named "FEVER One," where each sentence is labeled based on its support or refutation of the claim.

➔ Some FEVER claims need multiple statements as evidence, but our system focuses on claims that can be supported or refuted with a single sentence, avoiding complexities in handling multiple evidence statements.

- Decisions about each sentence are aggregated to decide the overall claim classification. If any sentence supports the claim, the claim is classified as supported. If any sentence refutes the claim without support, the claim is classified as refuted. If no evidence supports or refutes, the claim is marked as lacking information.

- To handle pronoun ambiguity, the system is made aware of the Wikipedia page title, providing contextual information. Adding titles to sentences significantly improves system performance, making it more accurate in assessing the claim.

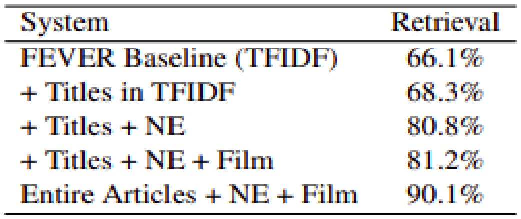
- The system naturally handles out-of-vocabulary words, especially names and terms, without additional modifications, providing an advantage over other models.

- Various strategies, such as incorporating titles, named entities, and film-related terms, progressively improve evidence retrieval accuracy. Retrieving entire articles, along with named entities and film-related terms, achieves the highest accuracy.

- The FEVER baseline system uses the Enhanced Sequential Inference Model (ESIM), which is a neural network architecture commonly used for natural language inference tasks. To balance these data, class reweighting is applied, ensuring fair evaluation and preventing bias. When the ESIM is replaced by the transformer network in our case, the transformer exhibits superior performance, eliminating the need for class reweighting and showcasing its effectiveness in intricate entailment tasks on the FEVER dataset.

Evidence Retrieval Strategy

Commencing with a baseline retrieval system that extracts sentences from the top five TFIDF-ranked articles, the study progressively refines this process. Simple yet effective modifications, such as adding titles to premise statements during TF IDF calculation, contribute to a modest increase in evidence retrieval to 68.3%. However, the adoption of more sophisticated strategies, including named entity recognition (NER) and film-related queries, substantially elevates the evidence retrieval rate to 81.2%. The most transformative adjustment involves eliminating TFIDF sentence ranking, enabling the retrieval of entire articles. This leads to a remarkable evidence retrieval rate of 90.1%, showcasing the system's adaptability to handle a diverse pool of 2.6 million statements for classification. These rates can be found in the table below.



System	Retrieval
FEVER Baseline (TFIDF)	66.1%
+ Titles in TFIDF	68.3%
+ Titles + NE	80.8%
+ Titles + NE + Film	81.2%
Entire Articles + NE + Film	90.1%

Table 3: Percentage of evidence retrieved from first half of development set. Single-evidence claims only.

Limitations

We recognized some limitations in the system we used and we present them below.

- **Dependency on Pretrained Models**: The system's performance relies on the quality of the pretrained models that are used.
- **Contextual Restrictions:** The scalability of generating NEI claims is limited since it necessitates more context from the FEVER dataset.
- **Named Entity Sensitivity Recognition**: The accuracy of NER affects the accuracy of claim generating.

We should also say that we couldn’t run the last part of the fact verification problem due to many errors we could not resolve. We read everything from the paper of the code and we discussed and understood the evaluation metrics that were used, as well as the difference of the models used with the baseline models of the FEVER dataset.

Topics Covered

During the making of project we came upon a wide range of topics covered in class, some of those are the following:

- One of the first contents we saw in our lectures was **Term frequencies** and **TF-IDF cosine similarity.** Although we did not explicitly use this, it was necessary for the creation of the initial FEVER dataset that we took as input.
- When extracting entities from sentences (our NER module), we make use of the **Stanza NLP** library. This is also used when generating NEI claims from <q,a> pairs.
- We used **Transformers** for the generation of Question-Answer pairs implemented with **BART**, which is a transformer-based model. We also use **BERT** and **RoBERTa-large** for the fact verification problem and the fine-tuning of the FEVER-training set.
- As with most training models, when it comes to evaluating results these are measured with **Precision**, important to assess the system's accuracy in identifying true positive claims, **Recall**, that ensures that the system does not miss relevant claims, **F1-score,** a balance between precision and recall, and other metrics related to (not exclusively) information retrieval.
- We also came across some definitions that are used in the baseline system of the FEVER dataset. The first one is **multi-layer perceptron (MLP)** and the second one is **decomposable attention (DA)**, both used for textual entailment recognition.

Bibliography

Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a Large-scale Dataset for Fact Extraction and VERification. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.

Dorottya Demszky, Kelvin Guu, and Percy Liang. 2018. Transforming question answering datasets into natural language inference datasets. arXiv.org.

Pan, L., Chen, W., Xiong, W., Kan, M., & Wang, W. Y. (2021b). Zero-shot Fact Verification by Claim Generation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*

Christopher Malon. 2018. Team Papelo: Transformer Networks at FEVER. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 109–113, Brussels, Belgium. Association for Computational Linguistics.