AUTOMATED CLAIM VERIFICATION USING TRANSFORMER MODELS AND THE FEVER DATASET

Mihika Rao (xsw5kn), Nina Chinnam (fhs9af)

1 INTRODUCTION AND MOTIVATION

Misinformation spreads faster than ever before in today's digital age, fueled by social media, open publishing platforms, and the ease of content sharing. While fact-checking platforms exist to address this challenge, the manual nature of traditional fact-checking makes it quite time consuming, labor intensive, and often reactive. This creates a widening gap between the speed at which misinformation spreads and the capacity to verify claims effectively.

Recent advances in Natural Langauge Processing (NLP) and the development of large-scale language models (LLMs) such as T5 and BART offer ways to automate parts of this verification pipeline. These models, trained on vast amounts of text data, have demonstrated impressive zero-shot and few-shot reasoning abilities, including recognizing entailment and generating structured outputs from free text.

The motivation behind this work is to explore whether such models can be leveraged to build an automated, end-to-end fact-checking pipeline where we ant to answer if claims can be automatically verified as supported, contradicted, not verifiable. Our goal is to develop a system that accepts free-form user input, extracts candidate claims, retrieves relevant evidence using external knowledge sources like Wikipedia, and performs claim verification using transformer-based models.

This work builds on established datasets such as FEVER, which provides a benchmark for evaluating automated fact-checking systems. By combining retrieval methods, named entity recognition, and zero-shot Natural Langauge Inference (NLI) models, we propose a scalable approach to tackle misinformation verification.

2 BACKGROUND

Fact-checking in the field of Natural Language Processing (NLP) is a multi-step task that requires understanding claims, retrieving relevant evidence, and reasoning over that evidence to determine the truthfulness of the claims. Unlike the classification or summarization tasks, fact-checking needs external grounding, meaning that claims must be verified against authoritative knowledge bases, such as Wikipedia, news archives, or scientific literature.

One of the most influential resources in this space is the FEVER (Fact Extraction and VERification) dataset. FEVER contains over 185,000 human-annotated claims derived from Wikipedia, each labeled as Supported, Refuted, or Not Enough Information, along with associated evidence sentences. The dataset has become the main benchmark for evaluating fact-checking systems, encouraging the development of both retrieval and verification models. It emphasizes not only verifying claims but also explaining the verification with supporting evidence, making it highly relevant to real-world applications.

Recent advancements in pretrained transformer models have transformed the landscape of NLP. Models like Flan-T5 and BART-MNLI have shown impressive capabilities in text generation, question answering, and natural language inference (NLI) - the task of determining if a premise supports, contradicts, or is neutral to a given hypothesis. Flan-T5 is a text-to-text model that can be prompted to perform a wide range of tasks, including claim extraction from unstructured input. BART-MNLI is trained on the Multi-Genre Natural Language Inference (MNLI) dataset, provides a zero-shot framework for claim verification by evaluating the relationship between a claim and a piece of evidence.

3 METHODS

Our system is designed as an end-to-end fact-checking pipeline that takes in unstructured text and outputs a verification label for each extracted claim. The pipeline consists of three main stages: Claim Extraction, Evidence Retrieval, and Claim Verification.

1) Claim Extraction We leverage Flan-T5, a text-to-text transformer model, to extract factual claims from free-form user input. Given an input paragraph, we prompt Flan-T5 to generate a list of factual statements, one per line. These statements often contain compound claims; therefore, we use spaCy's sentence segmentation to further split each statement into atomic claims to improve verification granularity.

2) Evidence Retrieval For each extracted claim, we perform Named Entity Recognition (NER) using spaCy to identify the strongest search term, prioritizing entities of type PERSON, ORG, GPE, LOC, or EVENT. If no such entity is found, the entire claim is used as the search query. We then query Wikipedia usiing the wikipedia Python package, retrieving the top matching articles. From the top-ranked article that contains the search term, we extract the first 1000 characters of content to serve as evidence for verification. If no relevant article is found, we return no evidence.

3) Claim Verification We apply the BART-MNLI model, a pretrained zero-shot Natural Language Inference (NLI) model, to evaluate the relationship between each claim and its retrieved evidence. BART-MNLI predicts one of three labels: - Entailment (Supported) - Contradiction - Neutral (Not Verifiable)

We use a natural language hypothesis template to frame the claim-evidence relationship and obtain the predicted label with the highest confidence score.

4) Frontend Interface The entire pipeline is wrapped in an interactive Gradio interface, allowing users to:

1. Submit a piece of text.
2. Select the claim to verify.
3. See the attached evidence.
4. See the verification verdict and supporting evidence.

This interface supports a step-by-step workflow, enabling bothh end-users and researchers to engage with the system in an interpretable and interactive manner.

4 EXPERIMENTAL RESULTS

We applied our end-to-end fact-checking pipeline on a randomly selected set of 100 claims from the FEVER development set. The results below summarize the performance across the three classification labels: Supported, Contradicted, and Not Verifiable.

Our pipeline achieved an overall accuracy of 22 percent , correctly verifying 22 out of 100 claims when compared against the FEVER ground truth labels. While this demonstrates that the pipeline can correctly verify claims in nearly half of the cases without task-specific fine-tuning, it also highlights the challenges of relying solely on zero-shot models and simple evidence retrieval strategies.

We further break down the performance by computing recall for each class. For supported claims, there were 16 correct out of 35 with a recall of 45.7 percent. For contradicted claims, there were 20 correct out of 48 with a recall of 41.7 percent. For not verifiable claims, there were 8 correct out of 17 with a recall of 47.1 percent.

In addition to the quantitative results, we conducted a qualitative error analysis and identified two common failure modes:

1. **Over-abstention (31% "Not Verifiable" Predictions)** The pipeline tends to fall back to "Not Verifiable" even when evidence exists in the retrieved text, indicating limitations in the model's confidence calibration or reasoning abilities.

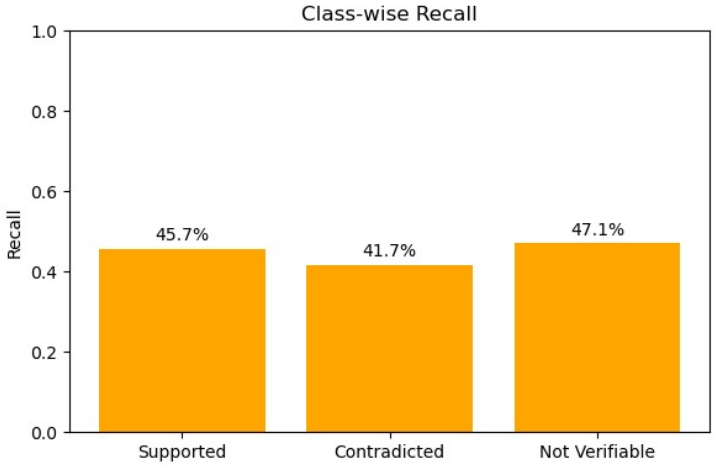

Figure 1: Class-Wise Recall

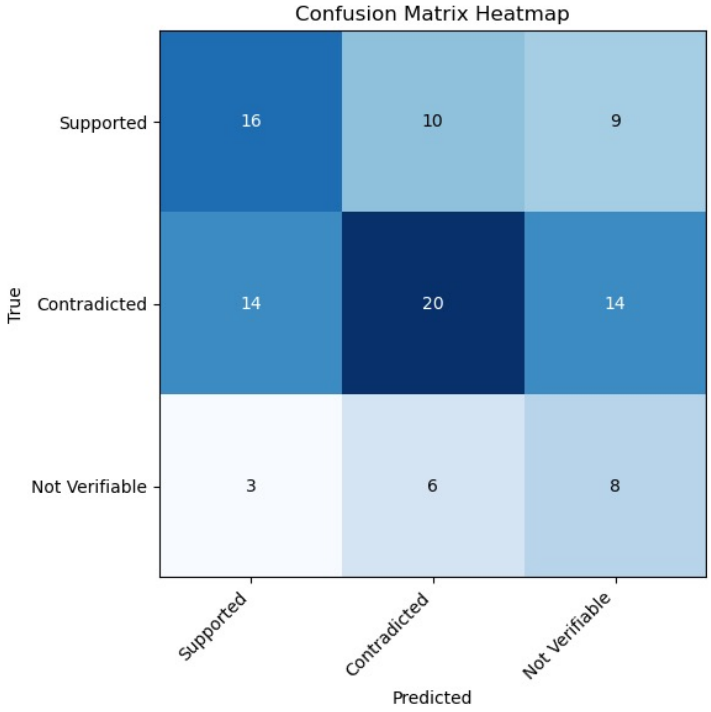

Figure 2: Confusion Matrix

2. **Weak Contradiction Detection (41.7% Recall)** The model struggles to correctly label false claims, potentially due to the complexity of negation and contradiction detection in natural language.

3. **Under-detection of Supported Claims (45.7% Recall)** The pipeline sometimes fails to recognize supportive evidence even when it is present, likely due to limitations in evidence retrieval or model inference.

5 CONCLUSION AND FUTURE WORK

In this project, we designed and implemented a fully automated, end-to-end fact-checking pipeline that ingests arbitrary user text, extracts individual factual claims, retrieves supporting evidence from Wikipedia, and verifies each claim with a zero-shot NLI model. By combining Flan-T5 for claim extraction, spaCy for sentence segmentation, the Wikipedia API for evidence retrieval, and a threshold-tuned BART-MNLI model for inference, we delivered a working prototype accessible through a Gradio interface that highlights verdicts and confidence scores back to the user.

When evaluated on a held-out subset of 100 claims from the FEVER v1.0 "labelled_dev" split, our pipeline achieved 44 perrcent overall accuracy. Class-wise recall rates were roughly balanced—Supported at 45.7 percent, Contradicted at 41.7 percent, and Not Verifiable at 47.1 percent—demonstrating that zero-shot transformers, even without any additional fine-tuning, can capture a substantial fraction of true facts and falsehoods when coupled with a simple retrieval strategy.

At the same time, the system exhibits systematic weaknesses. It tends to over-abstain, defaulting to "Not Verifiable" far too often, which points to both confidence-calibration issues in the NLI model and gaps in the retrieval step when relevant evidence isn't surfaced. Contradiction detection remains the hardest subtask, especially in the presence of negation or subtle paraphrasing, while support under-detection suggests that lexical variation between claim and evidence still trips up the model.

Looking forward, we identify several promising directions to bridge these gaps and push end-to-end accuracy toward—or beyond—the 70 percent range:

1. Semantic Retrieval: Replace or augment Wikipedia-API calls with a FAISS-indexed SBERT paragraph retriever to surface more relevant, paraphrase-robust evidence in sub-100 ms.

2. Multi-Snippet Aggregation: Retrieve the top k candidate snippets per claim and aggregate their entailment scores (e.g. via max- or mean-pooling) so that no single snippet can derail the final verdict.

3. Targeted Fine-Tuning and Data Augmentation: Expand the FEVER training set with adversarial examples—negations, numeric mismatches, entity swaps—and fine-tune our NLI model, optionally with class-weighted losses to boost contradiction performance.

4. Stronger NLI Backbones and Calibration: Experiment with newer MNLI checkpoints such as DeBERTa-v3-MNLI or cross-encoders, and perform a joint grid search over support/contradiction thresholds on a held-out subset to optimize precision, recall, and F_1.

5. Human-in-the-Loop and Explainability: Surface token- or sentence-level rationales (via attention, overlap scores, or saliency methods) for each verdict and enable quick manual correction of low-confidence cases—both to build user trust and to collect new training data.

6. Broader Domain Evaluation: Extend beyond Wikipedia/FEVER to news articles, scientific claims, or social media posts, benchmarking on complementary datasets and studying domain generalization.

In addition to the core pipeline, we built a lightweight Gradio interface that makes claim verification accessible to both technical and non-technical users: users paste any paragraph, extract and select individual claims from a dropdown, and immediately see the retrieved evidence alongside a color-coded verdict and confidence score. In future work we plan to enrich this UI by adding real-time progress indicators for retrieval and verification, a batch-mode workflow with CSV export for large-scale evaluation, and in-context highlighting of the exact sentence or tokens driving each verdict. We also aim to incorporate a "review mode" that surfaces low-confidence cases for human feedback—both to improve transparency and to gather high-quality labels for ongoing model refinement. Lastly, we'll explore responsive layouts and accessibility enhancements so that our tool can be used seamlessly on mobile devices and by users with varying needs.

By pursuing these enhancements—spanning retrieval, verification, data, and UX—we believe the pipeline can evolve into a robust, high-precision fact-checking assistant capable of real-time, large-scale misinformation mitigation tools and methods are openly shared so that the community can reproduce, benchmark, and build upon our findings.