Zero-Shot Fact Verification via Natural Logic and Large Language Models

Anonymous ACL submission

Abstract

The recent development of fact verification systems with natural logic has enhanced their explainability by aligning claims with evidence through set-theoretic operators, providing faithful justifications. Despite these advancements, such systems often rely on a large amount of training data annotated with natural logic. To address this issue, we propose a zero-shot method that utilizes the generalization capabilities of instruction-tuned large language models. To comprehensively assess the zero-shot capabilities of our method and other fact verification systems, we evaluate all models on both artificial and real-world claims, including datasets in Danish and Mandarin Chinese. We compare our method against other fact verification systems in two setups. First, in the *zero-shot generalization* setup, our approach outperforms other systems that were not specifically trained on natural logic data, achieving an average accuracy improvement of 8.61 points over the best-performing baseline. Second, in the *zero-shot transfer* setup, we demonstrate that current natural-logic-based systems do not generalize well to other domains. Our method performs better on all datasets with real-world claims compared to systems that were trained on datasets with artificial claims.

1 Introduction

In the context of fact-checking, fact verification (FV) is a process of verifying whether a textual hypothesis holds based on retrieved evidence. While many improvements have been made in this field due to the recent rapid growth in NLP (Mubashara et al., 2023; Guo et al., 2022; Nakov et al., 2021), FV systems often employ pipelines with black-box components that hide the underlying reasoning.

One line of research attempts to improve explainability with attention-based methods (Shu et al., 2019; Popat et al., 2018) and post-hock summarizations (Atanasova et al., 2020; Kotonya and Toni, 2020). However, these approaches do not provide *faithful justifications* — explanations that accurately reflect the model's decision-making process and the data it used (Jacovi and Goldberg, 2020). In contrast, systems such as NaturalLI (Angeli and Manning, 2014) and ProoFVer (Krishna et al., 2022) provide faithful justifications by expressing semantic relations between claim/evidence pairs. Modeling these logical relations and their aggregation explicitly with natural logic (NatLog) allows for handling phenomena such as double-negation and has resulted in more accurate and robust fact-checking systems.

However, a limitation of natural logic-based FV systems is the necessity for large amounts of training data annotated with entire natural logic proofs. For example, ProoFVer (Krishna et al., 2022) was trained on 145K instances artificially obtained from structured knowledge bases such as PPDB (Ganitkevitch et al., 2013) and Wikidata (Vrandečić and Krötzsch, 2014). While recent work (Aly et al., 2023) attempts to alleviate this issue by proposing a few-shot learning method trained on as few as 32 instances, human annotation of even a small number of proofs can be impractical and expensive, as it requires substantial linguistic knowledge and familiarity with natural logic. Moreover, few-shot systems require additional training data in order to generalize effectively to new domains, further increasing the costs.

To this end, we propose **Zero-NatVer**[1], a zero-shot fact verification approach for constructing natural logic proofs that leverages prompting and question-answering with instruction-tuned large language models (LLMs). Zero-NatVer's proof generation process is illustrated in Figure 1, consisting of a claim's chunking into smaller units of information, the alignment of claim chunks to relevant parts of the evidence, and the assignment

[1]Code is available at: https://github.com/TBD

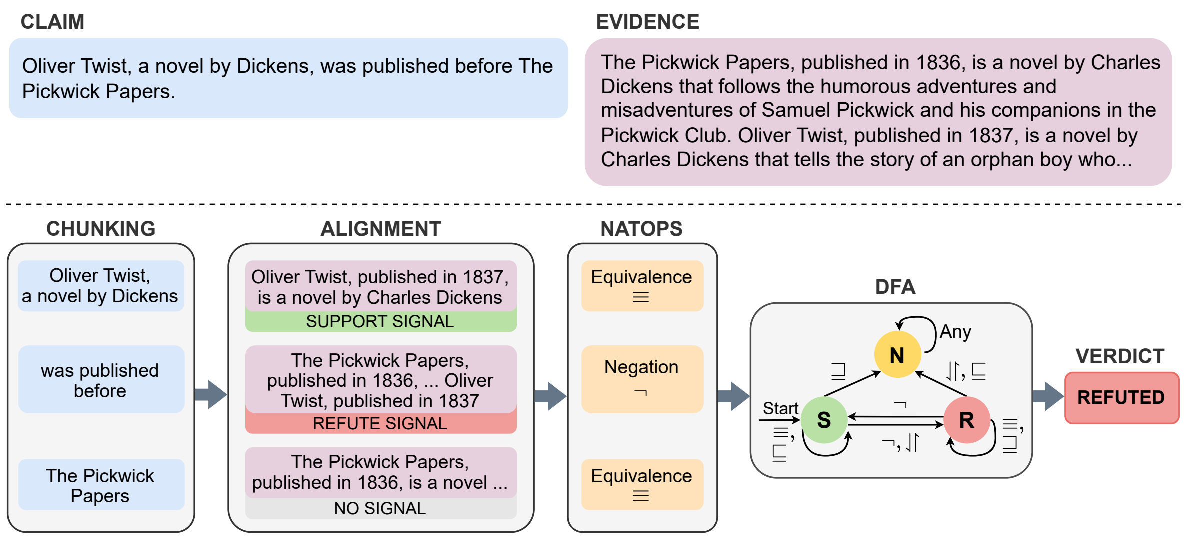

Figure 1: **Proof generation with natural logic in Zero-NatVer.** Initially, the claim and evidence texts are chunked and aligned. Zero-NatVer then assigns natural logic operators (NatOps), using a QA framework and alignment signals parsed from the previous step. This process produces a proof sequence comprising *(claim, evidence, NatOp)* triples. Lastly, NatOps act as transitions in the DFA, with the final state (here Refuted) determining the verdict.

of natural-logic operators to each aligned claim-evidence pair. The proofs are executed on a finite state automaton (DFA) as defined in natural logic inference. Contrary to previous works, our method uses a single language model for all stages of the proof generation pipeline. Zero-NatVer uses constrained decoding to prevent hallucinations during the chunking and alignment process. The alignment step further produces alignment justifications which are used in combination with QA ensembles to assign natural logic operators to claim-evidence pairs to reduce the variability of predictions and to account for missing context from the pair's restricted scope.

We evaluate our method on real-world and artificial FV datasets, including Climate-FEVER (Diggelmann et al., 2020), PubHealth (Kotonya and Toni, 2020), SciFact (Wadden et al., 2020), and Hover (Jiang et al., 2020). We also demonstrate that Zero-NatVer can generalize to non-English datasets by evaluating the system on the Danish dataset DanFever (Nørregaard and Derczynski, 2021) and the Mandarin Chinese dataset CHEF ((Hu et al., 2022)). In a zero-shot setup, where models have not been trained on any data labeled with natural logic, our approach outperforms all NatLog baselines by 8.61 accuracy points when averaged across all tested datasets. It is also competitive with the direct QA approach, where the model is prompted directly for an answer, achieving higher accuracy on all but two datasets and an average accuracy improvement of 3.16 points. Thus, our method, which is based on natural logic, provides both improved performance on unseen domains and explainability via faithful justifications.

2 Related Work

Natural logic (Van Benthem, 1986; Sanchez, 1991) and NaturalLI (Angeli and Manning, 2014), composes full inference proofs that operate directly on natural language, capable of expressing more complex logical relationships between claim and evidence, such as double-negation. Krishna et al. (2022) train natural logic inference systems for fact verification, achieving competitive performance while remaining faithful and more explainable than its entirely neural counterpart. While these neural-symbolic approaches require substantial training data to perform well, Aly et al. (2023) explore natural logic inference in a few-shot setting by casting natural logic operators into a question-answering framework, subsequently making use of the generalization capabilities of instruction-tuned language models. While our work also uses question answering to predict natural logic operators, we further address prediction calibration issues frequently encountered in a zero-shot setting (Kadavath et al., 2022; Jiang et al., 2023). Other neuro-symbolic reasoning systems for FV use simple logical rules to aggregate veracity information on a claim's components to provide simple faithful explanations (Stacey et al., 2022, 2023; Chen et al., 2022), however, these rules lack the expressiveness of natural logic and thus cannot inherently express more complex phenomena such as double negation.

Previous work on zero-shot FV is limited and largely relies on the generation of weakly supervised training samples and on knowledge of the target domain (Pan et al., 2021; Wright et al., 2022). Pan et al. (2023b) observe that typical FV systems fail when transferred to unseen domains in a zero-shot setting and propose a data augmentation technique to improve generalizability. However, none of the aforementioned zero-shot methods produces (faithful) explanations. In a few-shot setting, several recent works have explored the use of large language models that produce explanations alongside the verdict. Pan et al. (2023a) define a reasoning program consisting of a sequence of subtasks to verify complex claims. Yao et al. (2023) proposes chain-of-thought prompting complemented by action operations to support the model's reasoning and its explanation generation. Li et al. (2023) propose to edit rationales generated via chain-of-thought prompting by querying knowledge sources. Yet, in contrast to this work, these approaches still rely on in-context examples.

3 Zero-NatVer

Given a claim c and evidence sentences $e_1, e_2, ..., e_k \in E$, our system determines the veracity label y, which denotes whether the information from E supports c, refutes c, or whether there is not enough information to reach a verdict. Zero-NatVer obtains the verdict in four steps, executed by an instruction-tuned LLM.

In the first two steps, Zero-NatVer segments c into several chunks (Sec. 3.1) and aligns each such chunk with relevant information from E (Sec. 3.2). This process results in a sequence of l claim-evidence alignment pairs $A = a_1, a_2, ..., a_l$. As part of this alignment process, we also generate alignment explanations that are parsed for supporting/refuting signals. These signals are used in the third stage of the pipeline where Zero-NatVer determines semantic relations of aligned pairs in terms of natural logic. Thus, it generates a sequence of natural logic operators $O = o_1, o_2, ..., o_l$, which correspond to alignment pairs in A (Sec. 3.3). Finally, O is used in the last stage to traverse a deterministic finite automaton (DFA), which determines the claim's veracity. The following sections describe each step in more detail.

3.1 Chunking

"Oliver Twist, a novel by Dickens":
"Oliver Twist, published in 1837, is a novel by Charles Dickens"
+ **Supports** the expression

"was published before":
"The Pickwick Papers, published in 1836,... Oliver Twist, published in 1837"
+ **Refutes** the expression, as Oliver Twist was published after The Pickwick Papers

"The Pickwick Papers":
"The Pickwick Papers, published in 1836, is a novel by Charles Dickens"
+ Refers to the same entity

Figure 2: **Claim-evidence alignments with explanations.** The blue text indicates provided claim chunks, the purple text represents generated evidence alignments, and the black text denotes alignment explanations, which are parsed for signals in the NatOp assignment stage.

FV systems based on natural logic split claims into smaller, more manageable pieces, also called chunks (Krishna et al., 2022). These chunks, typically consisting of only a few words, represent a single atomic piece of information that can be independently verified and linked to relevant information in the evidence text.

We perform this task by prompting an LLM to *"Split the claim text into smaller chunks that can be individually fact-checked."* We then use constrained decoding to ensure the desired output format. Specifically, the model is allowed to either generate consecutive characters from the provided text or insert a special token (e.g., a newline character) to denote the start of a new chunk. This process is executed as follows:

1. The claim text c is pre-processed as a queue of tokens Q_C.
2. The decoding is prefixed with an initial phrase to encourage the generation of claim chunks.
3. The model is constrained to sample only one of two outputs - the next token from Q_C or a newline character.
4. Repeats step 3 until Q_C is empty (i.e., all claim tokens are consumed).

Given the constraints at each decoding step, the model cannot hallucinate new words, skip words, or alter information in the claim.

3.2 Alignment

In the second stage of the pipeline, each previously generated claim chunk is aligned with the corresponding information in the provided evidence sentences. We use an LLM to perform this alignment by prompting it with c, E, and all claim chunks (see details in Appendix D). Furthermore, we prompt the model to also generate alignment explanations for each generated alignment. 2 shows an example of the model's output.

To enforce the expected output format, we use constrained decoding, switching between three decoding modes: *claim*, *evidence*, and *alignment-explanation*. In the claim mode, we simply insert the chunk text, and no further text is generated. In the evidence mode, the model generates the alignment and is constrained so that it cannot use tokens that occur only in C and not in E. This constraint is meant to reduce hallucinations and prevent the model from aligning chunks with claim tokens. Lastly, the inference process is not constrained in the alignment-explanation mode because explanations are only searched for keywords and are not used in the following stages or as part of the proof.

Although constraint decoding helps mitigate hallucinations, it is important to note that the model could still hallucinate in evidence mode, as it is allowed to generate words not present in either C or E. Indeed, we analysed all alignments and found out that 12.4% of chunks contained at least one token absent from E. To solve this issue, we post-process the alignments and remove all text that does not form sequences of tokens in evidence sentences E. This post-processing step ensures that the alignment process is faithful and that only information from the evidence is used to verify the claim. Alternatively, we could constrain the decoding process to generate only tokens present in the evidence text. However, our empirical findings showed that this approach struggles in situations where it needs to combine two or more pieces of information that are not adjacent in the evidence text.

Lastly, the alignment explanations are parsed for supporting and refuting signals, which are used by the NatOp assigner. A simple keyword search was sufficient to effectively determine the signals while prioritizing precision over recall.

3.3 NatOp Assignment via QA Ensembles

Once the claim and evidence are aligned, the next step is to determine a single NatOp for each claim-evidence pair, which represents the semantic relation between the corresponding chunks.

We start by preparing the list of NatOp candidates for each alignment pair, considering five basic operators, as shown in Table 1. This process is guided by alignment signals from the previous stage, and we define the candidate lists as follows:

- For a supporting signal, we use operators that indicate the evidence chunk entails the claim chunk: $[\equiv, \sqsubseteq]$.
- For a negative signal, we use operators that indicate the claim chunk is not entailed by the information in the evidence chunk: $[\neg, \sqsupseteq, \downharpoonleft\upharpoonright]$.
- In case of no signal, the full set of NatOps is used: $[\equiv, \neg, \sqsubseteq, \sqsupseteq, \downharpoonleft\upharpoonright]$.

This process allows for transferring some global information from the aligner, which has access to the full claim and evidence texts, to the NatOp assigner, which only sees chunks and thus has limited knowledge. For example, in Figure 1, the aligner aligns *"was published before"* with corresponding years for each publication, describing the ordering of events. While this alignment is reasonable for a reader with access to the entire claim and evidence texts, it becomes challenging to determine its meaning if we only see the aligned sub-strings.

For each aligned pair, we then consider operators in the corresponding candidate lists, and this process is detailed in Figure 3. Similar to Aly et al. (2023), we treat these operators as relations that can be inferred via questions over claim-evidence spans. Thus, we prompt our model with *Yes/No* questions to determine whether a relation can be expressed

NatOp	Definition	Template Example
Equivalence ($\equiv$)	$x = y$	Is X a paraphrase of Y?
Forward Entailment ($\sqsubseteq$)	$x \subset y$	Given the premise X does the hypothesis Y hold?
Reverse Entailment ($\sqsupseteq$)	$x \supset y$	Does the expression Y entail X?
Negation ($\neg$)	$x \cap y = \varnothing \wedge x \cup y = U$	Is the phrase X a negation of Y?
Alternation ($\downharpoonleft\upharpoonright$)	$x \cap y = \varnothing \wedge x \cup y \neq U$	Does X exclude Y?

Table 1: Natural logic operators (NatOps) with set-theoretic definitions and template examples.

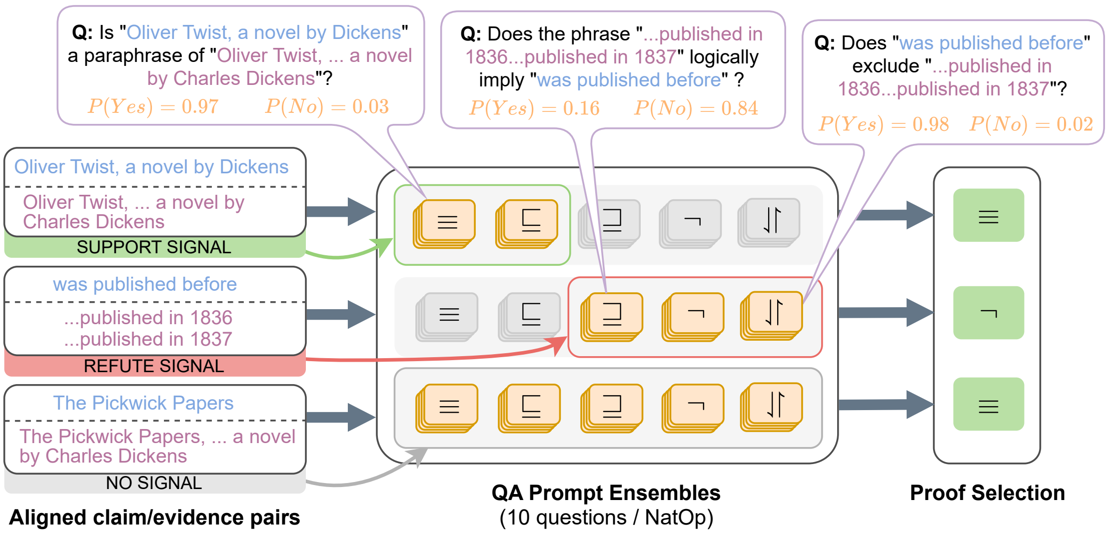

Figure 3: **Proof generation process of Zero-NatVer.** First, we utilize alignment signals, where available, to identify the set of potential NatOp candidates (represented by orange blocks). Next, we apply prompt ensembles and NatOp priority to select the final NatOp (depicted as green blocks).

by one of the NatOps. If none of these operators is successfully determined by the QA framework, we assign the independence operator $\#$, which implies that there is no semantic relation.

In order to reduce the variability of outcomes, we use a large number of *Yes/No* questions to prompt the model, thereby obtaining several micro-judgements per NatOp, which are then aggregated as a weighted average. In our experiments, we employ 10 templates for each NatOp. Rather than manually hand-crafting these question templates, we employ the LLM to generate them. Consequently, this approach allows for easy generation of additional templates as needed.

For a given claim-evidence alignment pair a and operator o, we compute a NatOp score $s_{o,a}$ as a weighted average over all micro-judgments:

$$s_{o,a} = \sum_{i=1}^{N} w_i \, \mathrm{QA}(\mathrm{Yes}|T_i, a) \qquad (1)$$

where T is a collection of prompt templates, and w represents confidence weights for each template, with $\sum_{i=1}^{N} w_i = 1$.

We compute w_i by iterating over the entire dataset in a single pass and capturing the log-likelihood scores for each template. For each instance, we always capture only the Yes/No option, which has the higher log-likelihood score (i.e., the option that the model favors more).

Using Equation 1, we then compile a list of NatOps candidates C, considering only $s_{o,a} > \alpha$, where α can be seen as a confidence threshold for the model. Since we are not using any validation data to determine hyper-parameters, we set $\alpha = 0.5$ as we are considering two output classes.

Due to the ambiguity of natural language and the complexity of alignments, it frequently occurs that $|C| > 1$. However, we want to minimize the chance of incorrectly choosing NatOps that leads to the *Not Enough Evidence* state, from which there are no outgoing transitions to other states. Thus, we use a NatOp priority approach and select from the operators in C in the following order: $[\equiv, \neg, \sqsubseteq, \sqsupseteq, \downharpoonleft\upharpoonright]$.We defined the NatOp order by considering the difficulty of each task. For instance, in a scenario where the candidate list C consists of equivalence ($\equiv$) and alternation ($\downharpoonleft\upharpoonright$), we postulate that identifying equivalence (i.e., assessing textual similarity) is a simpler task compared to identifying alternation (i.e., recognizing non-exhaustive exclusion). We decided on this order before our experiments and did not optimize this order.

4 Experimental Methodology

4.1 Zero-shot Setups

To better assess the zero-shot capabilities of our approach, we differentiate between two types of zero-shot setups– **zero-shot generalization** and **zero-shot transfer**. We define zero-shot generalization as a model's ability to handle entirely new tasks or domains it has not encountered during training. Conversely, zero-shot transfer refers to training a model on a specific task or dataset and subsequently applying it to a different but related task or dataset without further training. For example, consider a model trained on a broad spectrum of general data (e.g., BART, T5, or Llama) that did not include proofs with natural logic. Applying this model to FV with natural logic then exemplifies zero-shot generalization according to our definition. In contrast, if the same model is fine-tuned on a dataset annotated with natural logic proofs and then applied to perform FV with natural logic on a different dataset, this would be an instance of zero-shot transfer.

4.2 Datasets

Previous studies on NLI-based FV models have primarily focused on evaluating performance using artificial claims from FEVER-like datasets (Krishna et al., 2022; Aly et al., 2023; Chen et al., 2023). However, these datasets typically encompass only general topics, and artificial claims tend to be structurally simple. To achieve a more comprehensive assessment of zero-shot capabilities, we have evaluated our models on both artificial and natural claims, including non-English datasets.

For artificial claims, we evaluated models on claims from the multi-hop dataset Hover (Jiang et al., 2020) and the Danish dataset DanFEVER (Nørregaard and Derczynski, 2021). For real-world claims, we included English datasets Climate-FEVER (Diggelmann et al., 2020), PubHealth (Kotonya and Toni, 2020), and Scifact (Wadden et al., 2020), as well as the Chinese dataset CHEF (Hu et al., 2022). For datasets that provide knowledge bases for retrieval, we used BM25 (Robertson and Walker, 1994) to retrieve evidence. Further details are provided in Appendix A.

4.3 Baselines

Our natural-logic-based baselines consist of ProoFVer (Krishna et al., 2022) and QA-NatVer (Aly et al., 2023). We always try to use the largest possible backbone LLMs to make our results more comparable. However, both baseline models have specific limitations given by their current implementation.

ProoFVer currently supports only models from the Fairseq1 toolkit[2], and the largest supported model is BART (Lewis et al., 2019). For zero-shot transfer setups, we use ProoFVer with BART trained on 145K FEVER instances. For non-English datasets, we have use mBART (Liu et al., 2020) instead.

QA-NatVer can use larger LLMs such as Flan-T5 (Chung et al., 2022), but its implementation currently only supports training for encoder-decoder model architectures. Thus, we were unable to fine-tune QA-NatVer with Llama3 for zero-shot transfer experiments and used Flan-T5 trained on 64 instances instead. For experiments on DanFEVER, we used the mT0 (Muennighoff et al., 2022) backbone. The zero-shot generalization setup does not require any training, so we were able to use Llama3-8B for inference.

We also include results reported by Pan et al. (2023b) as an additional baseline for zero-shot transfer experiments. More details about our baselines can be found in Appendix B.

4.4 Implementation Details

We conducted our main experiments with the Llama3-8B model (AI@Meta, 2024). Crucially, we did not fine-tune the model on any specific dataset, and we did not tune any hyperparameters. The only exposure to fact-checking datasets was when we were designing our prompts. For this purpose, we used a separate dataset, Symmetric-Fever (Schuster et al., 2019). We selected a small subset of 100 claims and tested that our prompts generated responses in the desired format. For hyperparameters, we have adopted the recommendations of Perez et al. (2021) and did not rely on hyperparameters from prior works (details in Appendix C).

5 Results

Zero-shot Generalization We report the main results for zero-shot generalization in Table 2. We can see that Zero-NatVer outperforms other NatLog baselines across all datasets, covering artificial claims, real-world claims, and non-English datasets. Moreover, Zero-NatVer leverages a single multilingual model, offering broader applicability compared to QA-NatVer, whose chunker does not currently support Chinese. Consequently, we could not obtain results for the CHEF dataset.

Averaging results across all datasets, Zero-NatVer achieves an average accuracy of 59.25 points, outperforming ProoFVer by 19.42 accuracy points on average. Excluding the CHEF dataset, which QA-NatVer could not process, our system outperforms QA-NatVer by 17.62 accuracy

[2]https://github.com/facebookresearch/fairseq

System	Model	C-FEVER En		SciFact En		PubHealth En		Hover En		DanFEVER Da		CHEF Zh	
		F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
ProoFVer	BART/mBART	26.63	34.75	25.58	34.67	38.15	39.27	47.13	49.76	29.8	41.97	20.16	38.57
QA-NatVer	Flan-T5/mT0	22.20	36.86	23.56	40.67	44.42	48.73	35.65	50.85	35.68	37.05	-	-
QA-NatVer	Llama3-8B	32.6	36.5	37.18	43.67	63.66	68.79	49.95	54.93	48.92	55.35	-	-
Zero-NatVer	Llama3-8B	46.02	51.12	**54.58**	**58.33**	69.21	70.01	**60.26**	**60.27**	**53.9**	**62.55**	**47.94**	**53.2**
Direct-QA	Llama3-8B	**51.27**	**58.58**	52.76	57.00	**78.18**	**78.18**	55.34	57.00	52.77	61.7	19.5	24.04
Full Supervision	-	75.7	-	71.1	-	85.88	86.93	-	81.2	90.2	-	67.62	-

Table 2: **Zero-shot generalization results.** Macro-F1 and accuracy scores for systems that were **not** specifically trained on FV datasets. Where possible, we also report available SOTA results with fully-supervised models trained on in-domain data as a reference.

System	Model	Train size (FEVER)	C-FEVER En		SciFact En		PubHealth En		Hover En		DanFEVER Da		CHEF Zh	
			F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Pan et al.	BERT	800	40.60	-	50.71	-	60.06	-	-	-	-	-	-	-
ProoFVer	(m)BART	145K	40.70	43.35	45.57	49.16	57.78	61.22	57.08	57.89	36.12	55.22	20.18	37.72
QA-NatVer	Flan-T5/mT0	64	44.74	47.43	52.02	56.67	61.8	61.8	**70.27**	**70.5**	**63.64**	**68.41**	-	-
Zero-NatVer	Llama3-8B	None	**46.02**	**51.12**	**54.58**	**58.33**	**69.21**	**70.01**	60.26	60.27	53.9	62.55	**47.94**	**53.2**

Table 3: **Zero-shot transfer results.** Macro-F1 and accuracy scores for systems trained on the FEVER dataset. For each system, we report the provided language model and the size of the training data. Results from Pan et al. (2013) do not include accuracy scores and results for some of the datasets.

points when QA-NatVer utilizes the Flan-T5 backbone and by 8.61 points when it employs the Llama3-8B backbone.

We also reported SOTA results for each dataset to highlight the performance gap between models fully supervised on in-domain data and zero-shot approaches. The reported metrics, which include F1 and Accuracy scores where available, represent the best results to our knowledge.

Our results show that Zero-NatVer moves towards closing this gap while maintaining the significant advantage of utilizing a single model that does not require fine-tuning. In contrast, the results from SOTA involve six different models, each specifically fine-tuned to a particular dataset.

Direct-QA Table 2 also reports results for the Direct-QA setup, in which the Llama3 model was prompted to directly assign a verdict (i.e., Supported, Refuted, Not Enough Information) based on the provided claim and evidence texts. See Listing 3 for prompting details.

Zero-NatVer outperforms Direct-QA on all but two datasets, demonstrating its competitive performance while improving the model's explainability via generated proofs. Additionally, the results for Direct-QA might be overly optimistic. Given that Llama3 was trained on 15 trillion tokens, it is likely that some of the datasets were included in its training data. Since Zero-NatVer does not use Llama3 to directly predict the verdicts and the final verdict is derived from other tasks, its performance is likely to be more representative.

Zero-shot Transfer We report the main results for zero-shot transfer in Table 3. Zero-NatVer consistently outperforms both ProoFver and the results reported by Pan et al. (2023b) across all datasets, despite these baselines being trained on NatLog data and ProoFver's substantial training set of 145K instances. These findings highlight the robust generalization capabilities of Llama3, which our method effectively leverages.

When considering only datasets with natural claims (excluding CHEF), Zero-NatVer outperforms QA-NatVer by an average of 4.52 accuracy points. This indicates that while NatLog baselines trained on FEVER data generalize effectively to similar domains like Hover and DanFEVER (both predominantly featuring artificial claims from Wikipedia), their performance does not extend well to other domains. Therefore, in practical applications, it may be more advantageous to allocate computational resources to more powerful language models rather than training smaller models.

Ensemble size To assess the impact of the prompt ensemble size (Section 3.3), we run an experiment measuring performance for various en-

System	C-FEVER		SciFact		PubHealth		Hover		DanFEVER		CHEF	
	En		En		En		En		Da		Zh	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Zero-NatVer	46.02	51.12	54.58	58.33	69.21	70.01	60.26	60.27	53.9	62.55	47.94	53.2
- weighted templates	45.72	50.40	54.28	58.00	68.51	69.30	60.22	60.22	53.93	62.39	47.10	52.20
- QA templates	40.60	49.89	46.49	52.00	68.20	69.20	57.17	57.50	41.44	48.67	45.39	50.50
- constrained decoding	41.85	45.69	52.65	57.00	65.26	66.46	59.26	59.30	48.9	57.55	48.68	53.91
- alignment signals	40.62	43.66	52.27	55.00	54.94	55.22	58.72	58.73	48.15	52.91	43.27	49.22

Table 4: **Ablation study of Zero-NatVer.**

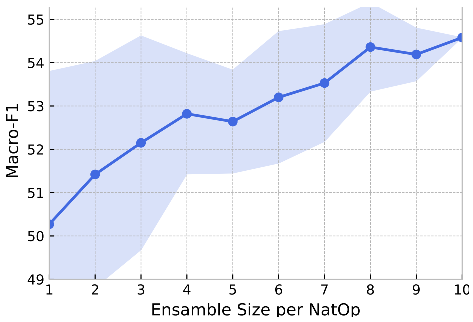

Figure 4: The averaged Macro-F1 scores for different ensemble sizes, calculated from 20 independent runs.

semble sizes. For each measured ensemble size S, we randomly sample S prompts for each NatOp from our prompt bank. We repeat this process 20 times and report means and standard deviations for each ensemble size in Figure 4.

The results indicate that the size of prompt ensembles significantly influences the variability of outcomes. When using only one question per NatOp and sampling different prompts, we obtain Macro-F1 scores with a standard deviation of 3.53 points. However, an ensemble of just four prompts significantly reduces this variation by more than half. Additionally, the performance consistently improves as the ensemble size increases.

	Macro-F1	Accuracy
Llama2-7B	20.57	41.67
Llama2-13B	30.96	42.16
Llama2-70B	57.47	60.33
Llama3-8B	54.58	58.33
GPT-3.5-Turbo	49.21	53.00

Table 5: SciFact results for LLMs of various sizes.

Model size Table 5 compares the performance of our method across different sizes and versions of Llama models, demonstrating a significant improvement as the model scales up. We also evaluated our method using the proprietary model ChatGPT-3.5 (OpenAI, 2023). Although ChatGPT-3.5 is allegedly larger than Llama3-8B, our method achieved better performance. This discrepancy may be attributed to API limitations, which prevented us from using constrained decoding and weighted prompting (see Appendix D for prompting details).

Ablation Study As reported in Table 4, we also perform four ablation studies to assess the importance of individual components in Zero-NatVer. First, we assess the performance without using weighted ensemble prompts and observe a slight decline of 0.49 accuracy points on average. Second, we ablate our method by omitting prompt ensembles and using a single randomly sampled prompt instead. We observe a substantial drop in performance of 4.62 accuracy points, which agrees with our previous findings regarding ensemble sizes. Third, we ablate Zero-NatVer by using unconstrained generation in decoding, observing an average accuracy drop of 2.6 points. Last, we ablate our method by removing alignment signals, observing a substantial drop of 6.79 average accuracy points.

6 Conclusion

We have presented Zero-NatVer, a zero-shot fact verification method grounded in natural logic. Our method leverages the generalization capabilities of instruction-tuned LLMs and generates faithful justifications for proofs without relying on training data annotated with natural logic. We have evaluated Zero-NatVer in two zero-shot setups, outperforming our baselines on most datasets. The ablation study shows the importance of individual design choices, and our comparison with the direct non-NatLog approach shows that natural logic provides competitive performance while providing explainability via faithful justifications. We hope that the methods and analyses presented here enable further progress toward improving the efficiency and explainability of fact verification systems.

Limitations

Natural logic is useful for explainability but is less expressive than semantic parsing methods such as lambda calculus (Zettlemoyer and Collins, 2005). This paper doesn't address natural logic's limitations. Furthermore, our method generates proofs, which are meant to be processed by the DFA from left to right. Nevertheless, natural logic-based inference is not constrained to such execution.

Ethics Statement

Intended Use and Misuse Potential. Our models can potentially captivate a wider audience and significantly reduce the workload for human fact-checkers. Nevertheless, it is crucial to acknowledge the possibility of their exploitation by malicious actors. As such, we strongly advise researchers to approach them with caution.

Accuracy and Infallibility. Our approach improves the clarity of FV models, enabling individuals to make better-informed decisions about trusting these models and their assessments. However, it is crucial for users to remain critical while interpreting the results of these systems and not mistake explainability for accuracy. We clarify that our evaluations do not determine the factual accuracy of a statement in the real world; instead, we use sources like Wikipedia as the basis for evidence. Wikipedia is a great collaborative resource, yet it has mistakes and noise of its own, similar to any encyclopedia or knowledge source. Therefore, we advise against using our verification system to make definitive judgments about the veracity of the assessed claims, meaning it should not be relied upon as an infallible source of truth.

References

AI@Meta. 2024. Llama 3 model card.

Rami Aly, Marek Strong, and Andreas Vlachos. 2023. Qa-natver: Question answering for natural logic-based fact verification. *arXiv preprint arXiv:2310.14198*.

Gabor Angeli and Christopher D Manning. 2014. Naturalli: Natural logic inference for common sense reasoning. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 534–545.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. Generating fact checking explanations. *arXiv preprint arXiv:2004.05773*.

Greg Brockman, Peter Welinder, Mira Murati, and OpenAI. 2020. Openai: Openai api. `https://openai.com/blog/openai-api`.

Jiangjie Chen, Qiaoben Bao, Changzhi Sun, Xinbo Zhang, Jiaze Chen, Hao Zhou, Yanghua Xiao, and Lei Li. 2022. Loren: Logic-regularized reasoning for interpretable fact verification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10482–10491.

Jiangjie Chen, Rui Xu, Wenxuan Zeng, Changzhi Sun, Lei Li, and Yanghua Xiao. 2023. Converge to the truth: Factual error correction via iterative constrained editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12616–12625.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904*.

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. Climate-fever: A dataset for verification of real-world climate claims. *arXiv preprint arXiv:2012.00614*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Xuming Hu, Zhijiang Guo, Guanyu Wu, Aiwei Liu, Lijie Wen, and Philip S Yu. 2022. Chef: A pilot chinese dataset for evidence-based fact-checking. *arXiv preprint arXiv:2206.11863*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*.

Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. 2023. Calibrating language models via augmented prompt ensembles.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. 2020. Hover: A dataset for many-hop fact extraction and claim verification. *arXiv preprint arXiv:2011.03088*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. *arXiv preprint arXiv:2010.09926*.

Amrith Krishna, Sebastian Riedel, and Andreas Vlachos. 2022. Proofver: Natural logic theorem proving for fact verification. *Transactions of the Association for Computational Linguistics*, 10:1013–1030.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Lidong Bing, Shafiq Joty, and Soujanya Poria. 2023. Chain of knowledge: A framework for grounding large language models with structured knowledge bases. *arXiv preprint arXiv:2305.13269*.

Bill Yuchen Lin, Kangmin Tan, Chris Miller, Beiwen Tian, and Xiang Ren. 2022. Unsupervised cross-task generalization via retrieval augmentation. *Advances in Neural Information Processing Systems*, 35:22003–22017.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Akhtar Mubashara, Schlichtkrull Michael, Guo Zhijiang, Cocarascu Oana, Simperl Elena, and Vlachos Andreas. 2023. Multimodal automated fact-checking: A survey. *arXiv preprint arXiv:2305.13507*.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated fact-checking for assisting human fact-checkers. *arXiv preprint arXiv:2103.07769*.

Jeppe Nørregaard and Leon Derczynski. 2021. Danfever: claim verification dataset for danish. In *Proceedings of the 23rd Nordic conference on computational linguistics (NoDaLiDa)*, pages 422–428.

R OpenAI. 2023. Gpt-4 technical report. arxiv 2303.08774. *View in Article*, 2.

Liangming Pan, Wenhu Chen, Wenhan Xiong, Min-Yen Kan, and William Yang Wang. 2021. Zero-shot fact verification by claim generation. In *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)*, pages 476–483, Online. Association for Computational Linguistics.

Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. 2023a. Fact-checking complex claims with program-guided reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6981–7004, Toronto, Canada. Association for Computational Linguistics.

Liangming Pan, Yunxiang Zhang, and Min-Yen Kan. 2023b. Investigating zero-and few-shot generalization in fact verification. *arXiv preprint arXiv:2309.09444*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *Advances in neural information processing systems*, 34:11054–11070.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. Declare: Debunking fake news and false claims using evidence-aware deep learning. *arXiv preprint arXiv:1809.06416*.

Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 232–241. Springer.

Victor Sanchez. 1991. *Studies on natural logic and categorial grammar*. Ph.D. thesis, University of Amsterdam.

Tal Schuster, Darsh J Shah, Yun Jie Serene Yeo, Daniel Filizzola, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. *arXiv preprint arXiv:1908.05267*.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 395–405.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Oana-Maria Camburu, and Marek Rei. 2023. Logical reasoning for natural language inference using generated facts as atoms. *arXiv preprint arXiv:2305.13214*.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, and Marek Rei. 2022. Logical reasoning with span-level predictions for interpretable and robust NLI models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3809–3823, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Johan Van Benthem. 1986. *Natural Logic*, pages 109–119. Springer Netherlands, Dordrecht.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. *arXiv preprint arXiv:2004.14974*.

Dustin Wright, David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Isabelle Augenstein, and Lucy Lu Wang. 2022. Generating scientific claims for zero-shot scientific fact checking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2448–2460, Dublin, Ireland. Association for Computational Linguistics.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Luke S. Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI'05, page 658–666, Arlington, Virginia, USA. AUAI Press.

A Dataset Processing

To effectively assess the zero-shot capabilities of FV systems, it is important to evaluate the performance on real-life claims and consider domains requiring various domain expertise. We evaluated all models on datasets covering natural claims and domains such as climate change, biomedical subjects, government healthcare policies, and scientific literature. We chose datasets that mainly focus on three-way classification, i.e., using three labels *Suppports*, *Refutes*, or *Not Enough Information*:

Climate-FEVER (Diggelmann et al., 2020) dataset comprises 1535 real-life climate change claims, each annotated with five evidence sentences retrieved from Wikipedia. Each evidence sentence was labeled by five human annotators as supporting, refuting, or inconclusive regarding the claim's veracity, resulting in 5 votes for each evidence sentence. These votes were then aggregated to micro-verdicts for each retrieved evidence sentence, and micro-verdicts were further aggregated to a single macro-label for the claim. In our data processing, we combined all evidence sentences into a single paragraph and paired them with the macro-label assessment. Besides the standard three labels, some claims in the datasets are labeled as *DISPUTED* if they are paired with both supporting and refuting micro-verdicts. Since our work focuses on three-label class prediction, we removed those 154 claims from the dataset.

PubHealth (Kotonya and Toni, 2020) is a dataset with natural claims in the public health domain. These claims are accompanied by evidence that requires subject matter expertise, along with expert explanations (judgments). The dataset contains four labels *True*, *False*, *Unproven*, and *Mixture*. However, the classes are heavily unbalanced and the labels *Unproven* and *Mixture* cover less than 10% of the data in total. Therefore, we use test set claims with only *True* and *False* labels, resulting in 987 claims paired with expert explanations as evidence.

SciFact (Wadden et al., 2020) is a dataset of expert-written scientific claims paired with evidence that was extracted from academic papers. We collect the claims with supporting and refuting rationale and construct claim-evidence pairs with *SUPPORT* and *REFUTE* labels. Claims lacking a specific rationale are categorized as *NEI*, and we

```
CLAIM: {C}
EVIDENCE: {E}

_____
Align the following claim expressions with relevant substrings from the evidence text:
* {CH–1}
* {CH–2}
...
* {CH–N}

The aligned substrings should either support the expression, refute it, or simply refer to the same entity.
Where possible, provide an explanation following each alignment.
If no relevant alignment exists, write "None".
```

Listing 1: Prompt template for the alignment task. Placeholders *{E}* and *{C}* get replaced by corresponding evidence and claim texts, respectively. Placeholders *{CH-1}* to *{CH-N}* get replaced by corresponding claim chunks, which were generated in the previous chunking step.

pair them with the entire abstract text. We evaluate our pipeline on a test set that consists of 300 claims.

Hover (Jiang et al., 2020) is an open-domain, multi-hop FV dataset, containing artificial claims built from the Wikipedia corpus. Its claims are labeled as either *SUPPORTED* and *NOT-SUPPORTED*. We use the development set, which consists of 4000 claims. In order to obtain evidence for all claims, we use the BM25 retriever (Robertson and Walker, 1994).

DanFEVER (Nørregaard and Derczynski, 2021) is a Danish dataset of counterfactual claims constructed from Danish Wikipedia. It consists of 6407 instances and provides gold evidence for *Supported* and *Refuted* claims. To obtain evidence for *NEI* claims, we use the BM25 retriever (Robertson and Walker, 1994).

CHEF (Hu et al., 2022) is a Chinese dataset of real-world claims. We use their development set, which consists of 703 claims.

B Baselines

ProoFVer (Krishna et al., 2022) is a seq2seq FV model that generates natural logic proofs as sequences of *(claim, evidence, NatOp)* triples. ProoFVer is based on GENRE (De Cao et al., 2020), an end-to-end entity linking model that was obtained by fine-tuning the BART language model (Lewis et al., 2019). ProoFVer was trained on a large collection of 145,449 claims from FEVER that were heuristically annotated with natural logic proofs.

QA-NatVer (Aly et al., 2023) is also based on natural logic but uses a question-answering framework to determine proofs. As a few-shot method, QA-NatVer was trained only on a small subset of FEVER data. It uses 64 training instances, which were further manually annotated with natural logic proofs.

QA-NatVer currently supports BART0 (Lin et al., 2022), Flan-T5 (Chung et al., 2022) and mT0 (Muennighoff et al., 2022) backbones.

Pan et al. Pan et al. (2023b) recently published an extensive analysis of zero-shot FV over 11 FV datasets. In their work, they experimented with different combinations of datasets for training and testing. While Pan et al. (2023b) consider their experiments as zero-shot generalization tasks, in our work, we consider them as zero-shot transfer because they train their models on other FV datasets. Their results show useful zero-shot baselines over most of our datasets, providing a comparison with FV models that are not based on natural logic.

C Models

Llama models For experiments with Llama3 (AI@Meta, 2024), we ran the 8B parameter model in 16-bit precision for inference. For experiments with Llama2, we locally ran the 7B, 13B, and 70B parameter models and used the GPTQ (Frantar et al., 2022) version of these models with 4-bit quantization to reduce computational requirements and accelerate inference.

Hyperparameters When decoding with Llama models, we did not tune any hyper-parameters and used the values described in Touvron et al. (2023).

Specifically, in the question-answering task for NatOPs, we set temperature to 1.0 and use nucleus sampling (Holtzman et al., 2019) with top-p set to 0.9. For all other tasks, we change temperature to 0.1.

Experimental Setup All experiments using Llama3 as the instruction-finetuned LLM were run on a machine with a single Quadro RTX 8000 with 49GB memory and 64GB RAM memory.

D Prompting

Listings1 show prompt templates for the evidence-rephrasing task, and the chunking and alignment task, respectively. These prompt templates were used for all experiments with Llama3 and ChatGPT models.

NatOp assignment Listing 2 shows the prompt templates used in the question-answering task for NatOps. Given a claim-evidence pair, we generated 10 distinct questions for each NatOp in separate prompts, replacing *X* with the claim text and *Y* with the evidence text. Additionally, we added the phrase *"Answer Yes or No."* at the end of each prompt to encourage the *Yes/No* output format. Lastly, we used the default system prompt *"You are a helpful assistant."* for all prompts.

ChatGPT We used OpenAI's API (Brockman et al., 2020) to query *gpt-3.5-turbo-1106* and used the same hyperparamteres as with Llama3 models. Due to the API limitations, we were unable to use constrained decoding for rephrasing, chunking, and alignment. Moreover, we could not use weighted prompt ensembles due to the inability to access the model's log-likelihood scores. Otherwise, we could replicate all the steps of our method with ChatGPT.

```
Equivalence
Is X a paraphrase of Y?
Are X and Y semantically equivalent in meaning?
Is the meaning of X effectively the same as Y?
Do X and Y function as synonyms or paraphrases of each other?
Does X serve as a paraphrase or an alternative expression for Y?
Are X and Y synonymous or nearly synonymous in meaning?
Do X and Y mean the same, without using external knowledge or assumptions?
Are X and Y semantically identical when considered independently of external knowledge?
Considering just X and Y, do these expressions have the same meaning?
Comparing X with Y, are they semantically equivalent based solely on their respective content?

Entailment
Given the premise Y does the hypothesis X hold?
Does the expression Y entail X?
Does the phrase Y logically imply X?
Is it true that if Y then X?
Is X a valid inference from Y?
Can X be inferred from the statement Y?
Given just the statements Y and X, does the first statement logically and necessarily imply the second without any external
information?
Is it true that the statement Y logically entails X based solely on the information within the statements?
Does Y imply X when only the information within these statements is considered?
Is it accurate to say that Y categorically entails X, without external interpretations?

Negation
Is the phrase X a negation of Y?
Do X and Y represent mutually exclusive states, where the presence of one negates the possibility of the other?
Is the relationship between X and Y binary, such that if X is true, Y must necessarily be false, and vice versa?
Do X and Y negate each other completely?
Are X and Y in a dichotomous relationship, where the existence of one implies the non-existence of the other?
Is there a mutually exclusive relationship between X and Y, indicating that only one can be true at any given time?
In the context of X and Y, does the affirmation of one mean the automatic negation of the other?
Do X and Y form a binary opposition, where one categorically negates the other?
Are X and Y opposites in such a way that they cannot be true simultaneously?
Is the relationship between X and Y characterized by a strict either/or dichotomy?

Alternation
Does X exclude Y?
Do X and Y represent distinct alternatives, but not the only possibilities in their category?
Are X and Y exclusively different without negating the existence of additional states or options?
Do X and Y denote exclusive but not exhaustive options within a larger set of possibilities?
In comparing X and Y, are they distinct yet not limiting the possibility of other variations or alternatives?
Are X and Y distinct entities or states that exclude each other without forming a complete, exhaustive set?
Are X and Y different entities or states, but not in a way that negates the possibility of other, different entities or states?
Are X and Y distinct entities or states that exclude each other without forming a complete, exhaustive set?
In comparing X and Y, are they exclusive in nature but not necessarily covering all possible alternatives?
Do X and Y define separate, non-intersecting options, while not encompassing all possible scenarios?
```

Listing 2: Template questions for determining NatOps.

```
Given the claim "{C}" and the evidence "{E}", determine if the evidence supports, contradicts, or is insufficient to conclude about
the claim.

Choices:
(A): Supports
(B): Refutes
(C): Not Enough Information
```

Listing 3: Prompt template for FV experiments in a direct multiple-choice setup. Placeholders *{E}* and *{C}* get replaced by corresponding texts.