Atomic Inference for NLI with Generated Facts as Atoms

Joe Stacey[1], Pasquale Minervini[2], Haim Dubossarsky[3], Oana-Maria Camburu[4], Marek Rei[1]
[1]Imperial College London, [2]University of Edinburgh,
[3]Queen Mary University of London, [4]University College London
{j.stacey20, marek.rei}@imperial.ac.uk
p.minervini@ed.ac.uk, h.dubossarsky@qmul.ac.uk, o.camburu@ucl.ac.uk

Abstract

With recent advances, neural models can achieve human-level performance on various natural language tasks. However, there are no guarantees that any explanations from these models are faithful, i.e. that they reflect the inner workings of the model. *Atomic inference* overcomes this issue, providing interpretable and faithful model decisions. This approach involves making predictions for different components (or *atoms*) of an instance, before using interpretable and deterministic rules to derive the overall prediction based on the individual atom-level predictions. We investigate the effectiveness of using LLM-generated facts as atoms, decomposing Natural Language Inference premises into lists of facts. While directly using generated facts in atomic inference systems can result in worse performance, with 1) a multi-stage fact generation process, and 2) a training regime that incorporates the facts, our fact-based method outperforms other approaches.[1]

1 Introduction

Current state-of-the-art models achieve impressive performance on various natural language understanding tasks. However, predictions from these models are not interpretable, and while existing methods can suggest plausible reasons for each prediction (Wiegreffe and Marasovic, 2021), there are no guarantees that these reasons are faithful to the underlying decision-making process of the model (Lyu et al., 2022; Atanasova et al., 2023). Despite the importance of inherently interpretable models for high-stakes decision-making (Rudin, 2019), few works on interpretability consider this type of model (Calderon and Reichart, 2024).

Motivated by this idea, we aim to introduce an inherently interpretable model that produces plausible and faithful explanations. This method involves decomposing an input into components (*atoms*) and making hard classification decisions independently for each atom. A sequence of interpretable and deterministic rules is then applied to derive the overall prediction based on the model decisions for these atoms. We refer to this approach as *atomic inference*, producing interpretable models that reveal the specific atom-level decisions responsible for each instance-level prediction.

Atomic inference methods are effective when underpinned by an appropriate choice of atoms, allowing models to independently make accurate predictions for each component part of an input. We investigate the effectiveness of using generated facts as our atoms. Specifically, we use an LLM to generate a comprehensive list of facts that summarises an input. This fact decomposition results in more atoms than a sentence segmentation, providing more fine-grained model interpretability. Moreover, we show that fact-based methods can considerably outperform existing methods that use either sentences or word spans as atoms.

We test our atomic inference methods on Natural Language Inference (NLI), a task that involves reasoning about the relationship between a premise and a hypothesis. This follows previous work with atomic methods, which often consider NLI (Schuster et al., 2022; Stacey et al., 2022; Chen et al., 2023) or tasks analogous to NLI (Glover et al., 2022; Laban et al., 2022; Kamoi et al., 2023; Zhang and Bansal, 2021; Yuan and Vlachos, 2023; Aly et al., 2023). To improve our fact-based models, we introduce different strategies to make the generated fact lists comprehensive, preventing important information from being missed during inference. We further experiment with an attention-based architecture that introduces the fact-generated atoms during training.

We describe our best performing system as FGLR (Fact-Generated Logical Reasoning), a method that achieves state-of-the-art results for

[1]https://github.com/joestacey/atomic_inference_anli/

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10188–10204
November 12-16, 2024

atomic inference, while also outperforming several large-scale LLMs.

2 Related Work

Atomic inference involves making discrete, atom-level predictions that are used to determine instance-level predictions (Stacey et al., 2022; Yuan and Vlachos, 2023), highlighting the specific atoms that are responsible for each model prediction[2]. This contrasts with atom-based methods that require soft atom-level predictions (Laban et al., 2022; Kamoi et al., 2023), or methods where the predictions for each atom also have access to other parts of the input (Wu et al., 2023; Chen et al., 2023; Feng et al., 2022).

Common choices of atoms include sentences (Schuster et al., 2022; Laban et al., 2022; Glover et al., 2022), word spans (Stacey et al., 2022; Aly et al., 2023; Braun and Kunz, 2024; Krishna et al., 2022), paragraphs (Glover et al., 2022; Laban et al., 2022), propositions (Chen et al., 2023), or semantic triples (Yuan and Vlachos, 2023). Recent work has further considered the decomposition of texts into lists of facts, using language models to generate fact lists that itemise the information present (Kamoi et al., 2023; Min et al., 2023). We consider the effectiveness of using generated facts for atomic inference, with models making hard entailment decisions about each fact.

Most atom-based methods either use existing NLI models to make atom-level predictions (Schuster et al., 2022; Glover et al., 2022; Laban et al., 2022; Kamoi et al., 2023), or provide additional atom-level annotations to be used for model training (Kamoi et al., 2023; Chen et al., 2023). We choose to take a different approach, integrating the fact-level decomposition into the model training process. Following Stacey et al. (2022), this approach teaches models to make accurate predictions for individual facts without requiring fact-level labels during training.

We provide a direct comparison of our system with the system proposed by Stacey et al. (2022), which segments NLI hypotheses into spans based on the presence of nouns. This span-level approach requires models to be trained with the atoms in-the-loop, enabling span-level predictions during inference. However, when using generated facts as atoms, we can compare the performance from training with the atoms in-the-loop to using a standard NLI model to make the atom-level predictions. Unlike Stacey et al. (2022), we also segment the NLI premise into atoms rather than the hypothesis, requiring a different framework for both training and inference. We also introduce a range of novel fact generation strategies to avoid missing information in our generated atoms, an issue that is avoided when segmenting instances into spans.

3 Method

3.1 Fact Generation

We define a *fact* as a statement representing a single piece of information. For each instance, we use GPT-3[3] (Brown et al., 2020) to generate a fact list that itemises all of the information contained within the premise (see Figure 1). To generate a list of facts, we provide the language model with the premise, followed by the instruction "List all the facts we explicitly know from the premise:".

We implement multiple fact-generation strategies with the aim of creating more comprehensive fact lists (see Figure 1), resulting in better performance. This involves (1) concatenating two independent lists of facts for each NLI premise, generated using different examples in the prompt, (2) asking a generator model to extend an existing fact list, and (3) generating facts that are also conditioned on a particular hypothesis. The hypothesis-conditioned facts are only generated for the test and validation data, so the model cannot access these facts during training. Providing these additional facts during training would require generating considerably more facts, with a substantially higher cost. Moreover, not providing the hypothesis-conditioned facts during training prevents models from learning from class-specific artifacts within the generated facts.

More details of the process, including an analysis of Figure 1, are included in Appendix C.1.

3.2 Model Architecture for Training

Our FGLR (Fact-Generated Logical Reasoning) model involves an attention-based architecture that is supervised to make predictions for individual facts while only using instance-level labels for training. This architecture has been used for token labelling (Rei and Søgaard, 2018; Pislar and Rei, 2020; Bujel et al., 2021) and NLI reasoning over

[2]For atomic inference, the possible labels for each atom do not need to align with the final task labels. For example, natural logic operators could be used as intermediate atom-level classes, similar to Aly et al. (2023).

[3]See Appendix A for more information

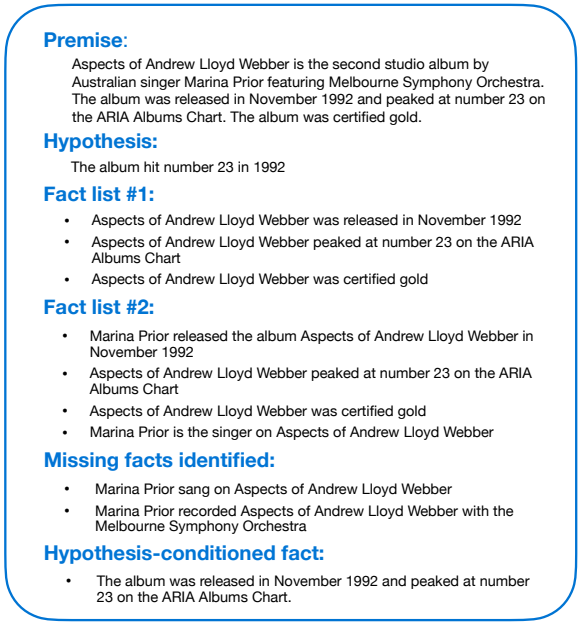

Figure 1: Generated fact-lists for a test example, including: 1) an initially generated fact-list, 2) a second generated fact-list that can be concatenated with the first list, 3) facts that an LLM identifies are missing from the original fact list, and 4) a generated fact that is conditioned on the hypothesis.

token spans (Stacey et al., 2022). Each individual premise fact i is encoded together with the full hypothesis using a pre-trained language model to create a representation R_{f_i}. Two separate linear layers (for detecting entailment and contradiction facts) are applied to this representation to create logits for entailment ($L_{e,i}$) and contradiction ($L_{c,i}$) for each fact i. To detect contradiction facts, unnormalized attention weights $\widetilde{a}_{c,i}$ are calculated as:

$$\widetilde{a}_{c,i} = \sigma(W_{c,2}(\tanh\left(W_{c,1}R_{f_i} + b_{c,1}\right)) + b_{c,2}) \quad (1)$$

with parameters $W_{c,1}$, $W_{c,2}$, $b_{c,1}$ and $b_{c,2}$, with the sigmoid σ bounding the value to a range between 0 and 1. These $\tilde{a}_{c,i}$ values are then normalised to create the attention distributions:

$$a_{c,i} = \frac{\widetilde{a}_{c,i}}{\sum_{k=1}^{m} \widetilde{a}_{c,k}} \quad (2)$$

An instance-level logit L_c is created from a weighted sum of the logits $L_{c,i}$, using the weights $a_{c,i}$. The logit $L_{c,i}$ represents a single premise fact i combined with the hypothesis, while L_c represents a combined score for all the facts in the premise (and therefore the whole instance). L_c is then supervised with a loss function to predict the target label y_c for each instance:

$$\mathcal{L}_c^{\text{Inst}} = (\sigma(W_{c,3} \times L_c + b_{c,3}) - y_c)^2 \quad (3)$$

In addition, the unnormalised attention values $\widetilde{a}_{c,i}$ are used in a fact-level loss, encouraging the model to assign more attention to facts in contradiction examples:

$$\mathcal{L}_c^{\text{Fact}} = (\max_i(\widetilde{a}_{c,i}) - y_c)^2 \quad (4)$$

This loss indirectly teaches the model to make fact-level decisions while only using instance-level labels. The value of y_c used in the supervision is determined by our training rules (see Section 3.3).

The same method is then used to detect entailment facts, with parameters $\{W_{e,j}, b_{e,j}\}_{j\in(1,2,3)}$, using the same representations R_{f_i} that are used in the contradiction fact detection. The different losses are then combined into

$$\mathcal{L} = \mathcal{L}_c^{\text{Fact}} + \mathcal{L}_e^{\text{Fact}} + \lambda(\mathcal{L}_c^{\text{Inst}} + \mathcal{L}_e^{\text{Inst}}) \quad (5)$$

using a hyper-parameter λ.

3.3 Rules for Training and Evaluation

The model architecture requires rules to determine the class labels during training, while also applying a second set of deterministic rules during inference. As the FGLR method decomposes the NLI premise into atoms rather than the hypothesis, the rules introduced by Stacey et al. (2022) are no longer applicable. We therefore require a new set of rules compatible with the model architecture. These rules are directly compared to the rules introduced by Stacey et al. (2022) in Appendix B.

The rules for training our method state that if an instance has a contradiction label, at least one model-generated fact must contradict the hypothesis. Similarly, if an instance does not have a contradiction label, then none of the model-generated facts contradict the hypothesis. The rules also state that if an example has an entailment label, then at least one of the model-generated facts must imply the hypothesis. This involves supervising our contradiction attention layer with $y_c = 0$ for entailment and neutral examples, and $y_c = 1$ for contradiction examples. For entailment examples, we supervise with $y_e = 1$, while for neutral examples, we use $y_e = 0$. We do not supervise the entailment attention layer for contradiction examples[4].

To apply this model at an instance level during inference, we make predictions only based on the values $\widetilde{a}_{c,i}$ and $\widetilde{a}_{e,i}$ for each fact i. If any $\widetilde{a}_{c,i}$ value

[4]We experimented with supervising $y_e = 0$ for contradiction examples but this marginally decreased accuracy.

is greater than 0.5 for any fact, then the instance is classified as a contradiction. Otherwise, if any $\widetilde{a}_{e,i}$ value is greater than 0.5, the instance is classified as entailment. Any instance not classified as either contradiction or entailment is predicted to be neutral.

4 Experiments

4.1 Datasets

We aim to introduce an atomic inference framework for challenging, multi-sentence NLI datasets where state-of-the-art models still have considerable room for improvement. As Adversarial NLI (ANLI, Nie et al., 2020) exemplifies this challenge, we focus our experimentation on this dataset. We additionally consider out-of-distribution performance for: ConTRoL (Liu et al., 2021), Recognizing Textual Entailment (RTE, Wang et al., 2018a) and Winograd NLI (WNLI, Wang et al., 2018b; Levesque, 2011). To avoid the baseline model needing to truncate premises, we filter ConTRoL to only include examples where the premise is < 2,000 characters.

4.2 Comparing Fact and Sentence Atomic Decompositions

We compare the performance of atomic inference systems when either using generated facts, or when directly segmenting the premise into sentences. Following previous work, we test performance when making fact-level (Kamoi et al., 2023) or sentence-level (Schuster et al., 2022; Laban et al., 2022) predictions using a standard NLI model, which we train on ANLI. We update these existing methods so that they follow our atomic inference rules for evaluation, describing these approaches as FactAI (for fact atoms), and SenAI (for sentence atoms). Both FactAI and SenAI involve the same baseline NLI model trained on ANLI, with the model either making predictions for each sentence (SenAI) or for each generated fact (FactAI)[5].

4.3 Training with Atoms in-the-loop

We show how fact-based methods perform better when trained with the fact atoms in-the-loop using our attention-based architecture. We additionally introduce a method of training with atoms in-the-loop using a sentence-level decomposition of the premise (which we call SenLR). This method conveniently avoids the need for a language model to generate facts, while also providing a strong comparison for our fact-based methods. Finally, we consider the performance of our fact-based model when applying alternative strategies to make the generated fact lists comprehensive. We describe our best performing system as FGLR, which involves training with a single fact list, before additionally including the hypothesis-conditioned facts during inference.

4.4 Baseline models

FactAI, SenAI, SenLR and FGLR are all model-agnostic methods that can be combined with a range of uninterpretable base models. We chose DeBERTa-base (He et al., 2021) due to its strong performance despite having relatively few parameters (<200m). This approach exploits the strengths of both LLMs and classification models, with LLMs proving to be effective at generating fact lists, but being prone to errors in fact-level entailment decisions (Min et al., 2023). We also provide further experimentation using BERT-base (Devlin et al., 2019) and DeBERTa-large models in Appendix D. We directly compare our models to SENTLI (Schuster et al., 2022)[6] and SLR-NLI (Stacey et al., 2022)[7], both atomic inference methods which we train on ANLI.

Finally, we compare our model performance to recent LLMs that were tested on ANLI by He et al. (2023), showing that models with our atom-level faithfulness guarantee can still reach or even exceed the performance of large-scale LLMs.

4.5 In-Distribution Results

For basic atomic inference systems, using generated facts as atoms does not outperform a sentence atom decomposition, with SenAI outperforming FactAI for each ANLI test-set (see SenAI vs FactAI in Table 1). However, when training with atoms in-the-loop and including the hypothesis-conditioned facts, the FGLR system outperforms all other atomic inference methods (see Table 1). Training with atoms in-the-loop considerably improves performance for both sentences and fact-generated atoms, however, the benefits from this approach are greatest when using the generated facts. While interpretable models usually need to sacrifice

[5]There are more facts per instance for ANLI compared to sentences, with 4.7 facts per instance on average compared to 3.0 sentences.

[6]In the case of SENTLI we only decompose the premise, as ANLI hypotheses do not require further decomposition.

[7]We exclude 0.02% of training examples due to the memory constraints of the SLR-NLI method, described in Appendix G

	In-distribution				Out-of-distribution			
	R1	R2	R3	ANLI-all	ConTRoL	RTE	WNLI	Int?
DeBERTa-base	71.2	54.0	51.7	58.5	53.7	85.0	59.6	✗
GPT-3.5-turbo[1]	68.5	54.4	55.9	59.4	-	-	-	✗
LLaMA2 70B[1]	69.1	54.8	54.1	59.0	-	-	-	✗
Mistral 7B[1]	55.5	43.0	42.5	46.7	-	-	-	✗
Span atoms:								
SLR-NLI[2]	65.5	47.8	47.1	53.0	48.9	82.3	56.3	✓
Sentence atoms:								
SENTLI[3]	69.5	53.5	51.3	57.7	52.3	82.0	**60.7**	✓
SenAI	69.3	53.5	51.6	57.7	50.0	81.7	60.6	✓
SenLR (ours)	**71.5**‡	**55.0**‡	**52.3**	**59.1**‡	**53.4**‡	**83.7**‡	53.8	✓
Fact atoms:								
FactAI	65.2	50.6	49.9	54.9	46.6	77.2	**74.6**	✓
FGLR (ours)	**71.8**‡	**56.1**‡	**55.3**‡	**60.7**‡	**49.1**‡	**80.8**‡	70.7	✓

Table 1: Model accuracy after training on ANLI. [1] represents few-shot CoT results reported by (He et al., 2023), while [2] and [3] are baselines recreated from Stacey et al. (2022) and Schuster et al. (2022) respectively using the DeBERTa base model. † represents results that are statistically better than the corresponding SenAI or FactAI baseline with $p < 0.05$, while ‡ represents results where $p < 0.01$, using bootstrapping statistical testing (Efron and Tibshirani, 1993). 'Int?' indicates whether the model is interpretable. All results are an average from 10 seeds.

some performance (Calderon and Reichart, 2024), we find that FGLR even outperforms very large generative models. In particular, the biggest advantage of FGLR compared to other atomic inference methods is the strong performance on ANLI round-3, suggesting that fact-generated atoms help most on challenging NLI examples.

In addition to our experimentation using DeBERTa-base, we provide results in our Appendix for implementing atomic methods with both DeBERTa-large and BERT models (see Table 2 and Table 3). We also provide a range of different ablation experiments in Appendix E to check that all the components of FGLR are indeed necessary and beneficial. These experiments show that just under half of the improvements between FactAI and FGLR are a result of the hypothesis-conditioned facts used during inference (Table 6).

4.6 Out-of-Distribution Results

We identify weaknesses in atomic inference systems when testing in out-of-distribution settings, a phenomena that has not been considered in previous work. Table 1 shows how the ANLI-trained atomic inference models perform worse than the non-interpretable DeBERTa base model for two of the three OOD datasets tested. However, we show that training with the atoms in-the-loop can help to mitigate this issue, considerably improving performance on both ConTRoL and RTE (see SenAI vs SenLR, and FactAI vs FGLR in Table 1).

5 Conclusion

We experiment with using LLM-generated facts as atoms in atomic inference systems, decomposing each NLI premise into a list of facts before making entailment predictions for each fact with the hypothesis. The instance-level predictions then depend entirely on the model's granular predictions about each fact. We find that using a standard NLI model to make predictions at a fact level results in worse performance than existing methods. However, when 1) including a multi-stage fact generation process, and 2) incorporating the generated facts during model training, our fact-based approach outperforms existing atomic inference methods. Our resulting FGLR model makes fact-level predictions and combines them with logical rules, without requiring fact-level labels during training. This results in high-performing, interpretable models that specify exactly which facts are responsible for each model prediction.

Limitations

To distinguish between the entailment and neutral classes, we predict the premise as entailing the hypothesis whenever one of the individual facts from the premise implies the entire hypothesis. This approach prevents models from performing multi-hop reasoning across different premise facts. While we find that reasoning across multiple facts is unnecessary for strong performance on ANLI, there may be other datasets where this would limit performance. We propose addressing this limitation in future work.

Our method relies on decomposing the NLI premise into facts (or *atoms*), determining the specific part of the input responsible for each model prediction. However, as ANLI consists of single-sentence hypotheses, no additional decomposition is required for the hypothesis. Further work would be needed to consider how to perform atomic inference over both a multi-sentence premise and a multi-sentence hypothesis when training with atoms in-the-loop.

Additionally, while our model provides interpretable decisions at an atom level, each atom-level decision itself is not interpretable. This method enables strong performance on NLI, while also providing faithfulness guarantees for the atom-level predictions.

Finally, we use GPT-3 to generate facts for each premise, which for this work cost ∼400 USD. As a result, we focused our experimentation on ANLI, while also including out-of-distribution evaluation on the RTE, WNLI, and ConTRoL datasets.

Acknowledgements

We would like to thank Greg Durrett, Rami Aly and Derek Chen for all their valuable feedback on this work.

Joe Stacey was supported by the Apple Scholars in AI/ML PhD fellowship. Oana-Maria Camburu was supported by a Leverhulme Early Career Fellowship. Pasquale was partially funded by the European Union's Horizon 2020 research and innovation programme under grant agreement no. 875160, ELIAI (The Edinburgh Laboratory for Integrated Artificial Intelligence), EPSRC (grant no. EP/W002876/1), an industry grant from Cisco, and a donation from Accenture LLP. Haim was supported by the Riksbankens Jubileumsfond (under reference number M21-0021, Change is Key! program).

References

Rami Aly, Marek Strong, and Andreas Vlachos. 2023. Qa-natver: Question answering for natural logic-based fact verification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 8376–8391. Association for Computational Linguistics.

Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. Faithfulness tests for natural language explanations. *ACL*.

Marc Braun and Jenny Kunz. 2024. A hypothesis-driven framework for the analysis of self-rationalising models.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Kamil Bujel, Helen Yannakoudakis, and Marek Rei. 2021. Zero-shot sequence labeling for transformer-based sentence classifiers. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 195–205, Online. Association for Computational Linguistics.

Nitay Calderon and Roi Reichart. 2024. On behalf of the stakeholders: Trends in nlp model interpretability in the era of llms.

Sihao Chen, Senaka Buthpitiya, Alex Fabrikant, Dan Roth, and Tal Schuster. 2023. PropSegmEnt: A large-scale corpus for proposition-level segmentation and entailment recognition. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8874–8893, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bradley Efron and R Tibshirani. 1993. An introduction to the bootstrap.

Yufei Feng, Xiaoyu Yang, Xiaodan Zhu, and Michael Greenspan. 2022. Neuro-symbolic natural logic with introspective revision for natural language inference.

Transactions of the Association for Computational Linguistics, 10:240–256.

John Glover, Federico Fancellu, Vasudevan Jagannathan, Matthew R. Gormley, and Thomas Schaaf. 2022. Revisiting text decomposition methods for NLI-based factuality scoring of summaries. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 97–105, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Xuanli He, Yuxiang Wu, Oana-Maria Camburu, Pasquale Minervini, and Pontus Stenetorp. 2023. Using natural language explanations to improve robustness of in-context learning for natural language inference.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. Wice: Real-world entailment for claims in wikipedia.

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.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Hector J. Levesque. 2011. The winograd schema challenge. In *Logical Formalizations of Commonsense Reasoning, Papers from the 2011 AAAI Spring Symposium, Technical Report SS-11-06, Stanford, California, USA, March 21-23, 2011*. AAAI.

Hanmeng Liu, Leyang Cui, Jian Liu, and Yue Zhang. 2021. Natural language inference in context - investigating contextual reasoning over long texts. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13388–13396. AAAI Press.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2022. Towards faithful model explanation in NLP: A survey. *CoRR*, abs/2209.11326.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Miruna Pislar and Marek Rei. 2020. Seeing both the forest and the trees: Multi-head attention for joint classification on different compositional levels. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3761–3775, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Marek Rei and Anders Søgaard. 2018. Zero-shot sequence labeling: Transferring knowledge from sentences to tokens. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 293–302. Association for Computational Linguistics.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.*, 1(5):206–215.

Tal Schuster, Sihao Chen, Senaka Buthpitiya, Alex Fabrikant, and Donald Metzler. 2022. Stretching sentence-pair NLI models to reason over long documents and clusters. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 394–412, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 3809–3823. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018a. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP:*

Analyzing and Interpreting Neural Networks for NLP, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Sarah Wiegreffe and Ana Marasovic. 2021. Teach me to explain: A review of datasets for explainable natural language processing. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Zijun Wu, Zi Xuan Zhang, Atharva Naik, Zhijian Mei, Mauajama Firdaus, and Lili Mou. 2023. Weakly supervised explainable phrasal reasoning with neural fuzzy logic. In *The Eleventh International Conference on Learning Representations*.

Zhangdie Yuan and Andreas Vlachos. 2023. Zero-shot fact-checking with semantic triples and knowledge graphs. *CoRR*, abs/2312.11785.

Shiyue Zhang and Mohit Bansal. 2021. Finding a balanced degree of automation for summary evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6617–6632, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Modelling Details

We perform the fact generation for our fact lists with text-curie-001, using the same model when extending these lists of facts. However, we used *text-davinci-003* when generating the facts conditioned on the hypothesis, finding that generating the facts conditioned on the hypothesis was a more difficult task. As further out-of-distribution experiments were conducted when these models were no longer available, facts were generated for the RTE and WNLI datasets using GPT-3.5-turbo (Brown et al., 2020; Ouyang et al., 2022).

For our base models, we used bert-base-uncased, deberta-v3-large, and deberta-v3-base, implemented from HuggingFace (Wolf et al., 2020). All statistical testing was performed using a bootstrapping hypothesis test (Efron and Tibshirani, 1993). A diagram describing FGLR can also be found in Figure 4.

B Logical Rules Comparison

We compare our training and evaluation rules to those presented by Stacey et al. (2022) (see Figure 2), comparing the different approaches when either segmenting the NLI premise or hypothesis. For Stacey et al. (2022), when decomposing the hypothesis into atoms, entailment is the default class (when no contradiction or neutral atoms are detected), whereas when decomposing the premise into atoms in this work, neutral is the default class (if no contradiction or entailment atoms are detected).

C Fact Generation Strategies

C.1 Method

To maximise the likelihood that all premise information is contained in our fact list, we investigate three different fact generation strategies: 1) concatenating two independent lists of facts to reduce the likelihood that key facts are missing (we note that it should not matter if some facts are repeated), 2) we extend a given fact list to identify potentially missing facts, and 3) we generate an additional fact conditioned on the hypothesis.

When using two independent lists of facts, we generate the fact list a second time using our generator model with different examples in the prompt. These two fact lists are then combined during inference. We initially experimented with also combining both sets of fact lists during training, but we found this resulted in marginally worse performance. In the example in Figure 1, information about the singer is missing in the first fact list, but is included in the second fact list. Therefore, by concatenating both fact lists, we provide a more comprehensive list of facts to support our model predictions. There should be no issue with similar, duplicated facts, as the atom-level predictions should be the same for facts with identical information.

Alternatively, we extend each fact list, using our generator model to generate additional facts to complement the initial facts already generated. In this case, the generator is presented with examples of incomplete fact lists in the prompt where the missing facts are then subsequently identified. This method aims to use our generator LLM to identify and remedy instances where key facts in the premise are missing. In Figure 1, this method successfully includes missing information about both the album singer and the Melbourne Symphony Orchestra.

Finally, we experiment with generating premise facts conditioned on the hypothesis. This involves generating an additional fact for each example, asking the model to provide a fact explicitly known from the premise that can be used to verify if the hypothesis is true. By asking the model to produce a single fact conditioned on the hypothesis, we encourage the model to include all the relevant information in a single fact rather than requiring multi-fact reasoning. This is the case for Figure 1, where the relevant information is condensed into a single fact that prevents the need for multi-fact reasoning.

For our fact-decomposition language model, four examples are provided in the prompt, making fact generation a few-shot task. As multiple hypotheses correspond to each premise in ANLI, not including the hypothesis in the prompt for the training data also substantially reduces the number of facts that need to be generated.

C.2 Results from Fact Generation Strategies

Out of the three fact generation strategies, we find better performance on the validation set when using hypothesis-conditioned facts (60.4% dev accuracy), compared to using a combined fact list (58.6% dev accuracy) and extending the fact lists (58.0% dev accuracy). We show the test performance of each of these systems in Table 4. The lower performance when extending a fact list can be explained by an increase in hallucinations with this strategy.

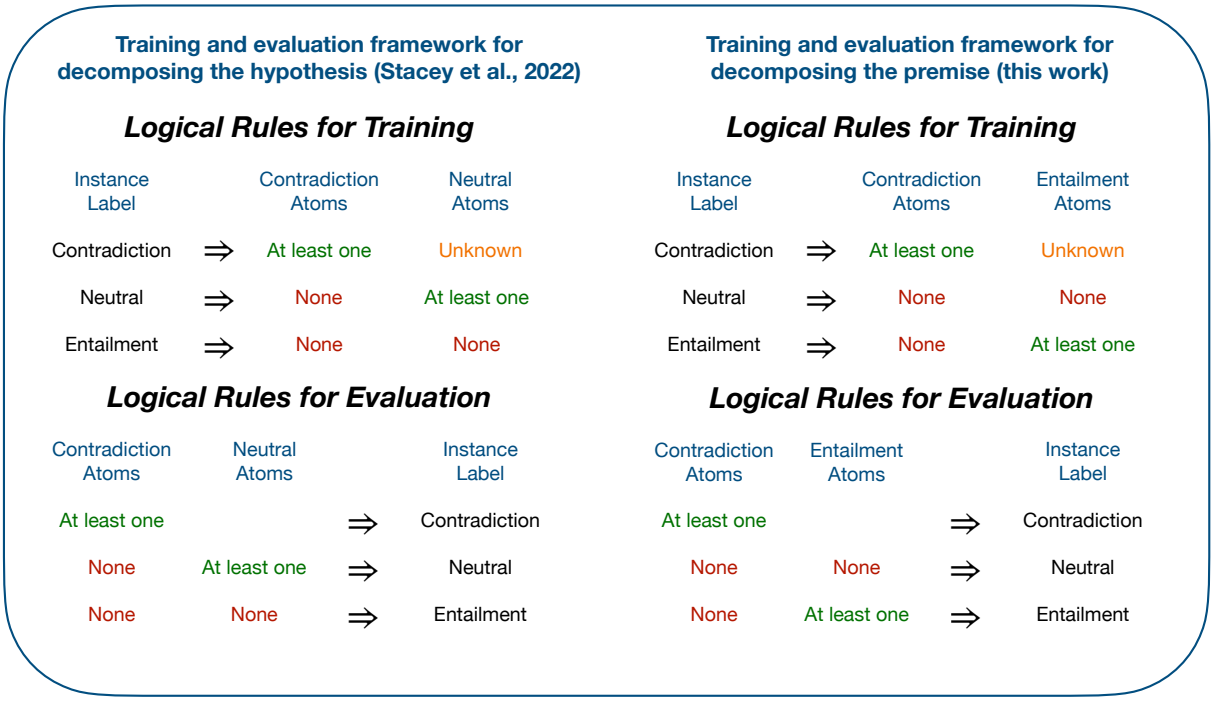

Figure 2: Inference and training framework from our work decomposing the NLI premise, compared to the rules used by Stacey et al. (2022) when decomposing the NLI hypothesis.

In Table 4 we additionally show performance from combining different fact-generation strategies, e.g. using combined fact lists and using hypothesis-conditioned facts, or extending the fact lists and using hypothesis-conditioned facts. These results show that combining these different strategies does not improve performance.

D Additional experimentation

In addition to the experiments performed in the main paper using DeBERTa-base, we perform additional experiments using DeBERTa-large (see Table 2) to consider the effectiveness of our method when applied to a high-performing base model with close to state-of-the-art results. We also experiment with a BERT-base model (see Table 3), to consider if the findings still apply to a worse-performing base model. When applied with a DeBERTa-large model, our FGLR method significantly outperforms the FactAI baseline model for each of the ANLI in-distribution test sets (R1, R2 and R3), and also for two of the out-of-distribution test sets (ConTRoL and RTE). We also see SenLR significantly outperforming the SenAI baseline for each of the in-distribution test sets, and also for ConTRoL. When using a BERT-base baseline, FGLR is significantly better than FactAI for each ANLI dataset and also for RTE. Accuracy from SenLR is significantly better than SenAI for ConTRoL and RTE, although there is little difference in-distribution compared to the baseline.

We perform additional experiments when training on the ConTRoL dataset. These experiments include out-of-distribution evaluation on our other NLI datasets (ANLI round 1, ANLI round 2, ANLI round 3, RTE and WNLI) - see Table 9. In this case, we find that the baseline model and FactAI do not converge (with the final model predicting only the entailment class), whereas FGLR performs better across each NLI dataset. As explained in Section 4.1, we reduce the ConTRoL data so that no premises exceed 2,000 characters. This avoids an unfair comparison where there is truncation required for the uninterpretable base model.

Finally, we also provide the standard deviations for each result in Table 1 in Table 10.

E Ablation studies

We perform an extensive range of ablation experiments to identify the specific aspects of FGLR that are responsible for its strong performance compared to the FactAI baseline (Table 4, Table 5, Table 6, Table 7 and Table 8).

First, we consider the performance of FGLR when there is either no sentence loss or fact loss (see Table 4). These results demonstrate the impor-

tance of the fact-level loss, while the instance-level loss is responsible for a small improvement in performance. As we aim to improve the performance of the interpretable atomic inference models, we include this small performance improvement.

We additionally experiment with using different strategies for generating the fact list for each example (see Table 4). This includes removing the hypothesis-conditioned facts, using our strategy of combining multiple fact lists instead of using the hypothesis-conditioned facts, and combining both strategies together (using the hypothesis-conditioned facts in addition to the combined fact list). The best approach (using the hypothesis-conditioned facts) was selected based on performance on the validation set.

To understand whether the facts contain additional information that the DeBERTa-base model does not have, we also try appending these facts to the uninterpretable DeBERTa-base model. However, we find that the model does not converge in this setting (see Table 5). This suggests that any comparison between our interpretable FGLR model and the uninterpretable DeBERTa-base model is likely to be a fair comparison.

We also consider the extent to which the FactAI baseline can be improved by including the hypothesis-conditioned facts (see Table 6). We find that improving the fact generation strategy for FactAI improves ANLI accuracy from 54.9% to 57.4%, almost half of the overall improvement from the complete FGLR system (60.7% accuracy). This supports our approach of training with atoms in-the-loop, in addition to our improved fact-generation strategies.

As we utilise GPT-3 to generate our fact lists, we additionally experiment with using GPT-3 to make entailment decisions for each hypothesis and premise fact pair (see Table 7). Specifically, we use a GPT-3.5-turbo model to do this in a few-shot setting (providing two examples to the model). Due to the poor performance, we concentrate our efforts on the better-performing DeBERTa models.

Finally, we provide a further experiment comparing the performance of the DeBERTa-base model to an adapted version of FGLR that does not use any atom decomposition, but instead only uses the full premise and hypothesis as inputs (See Table 8). We find this system provides marginally better performance than the DeBERTa-base baseline (59.3% compared to 58.4%), but is still substantially lower than the performance of the full FGLR system (60.7%).

F Examples of our FGLR system

We provide some examples to show the interpretability benefits of FGLR (Figure 3, Figure 5 and Figure 6). These examples are from the round 1 ANLI validation set, and consist of the generated fact-list in addition to the hypothesis-conditioned fact for each example. For Figure 3, we compare predictions of FGLR to predictions of FactAI, showing an example where FGLR makes better predictions at a fact level.

G Hyper-parameter Tuning and Baselines

For each model, we experiment with the following learning rates: 1×10^{-6} to 9×10^{-6} in increments of 1×10^{-6}, and 1×10^{-5} to 9×10^{-5} in increments of 1×10^{-5}. The base models performed best using learning rates of 6×10^{-5}, 4×10^{-5}, and 5×10^{-6}, for BERT-base, DeBERTa-base, and DeBERTa-large, respectively, while the best FGLR methods used lower learning rates of 5×10^{-6}, 7×10^{-6}, and 3×10^{-6}, respectively. For the λ value, we experiment with values of 0.1 to 1 in increments of 0.1, choosing a value of 0.9. Finally, we find marginally better performance if the FGLR encoder is initialised with the parameters of the fine-tuned base model. Our DeBERTa-base model consists of 184 million parameters, compared to 110 million for BERT and 304 million for DeBERTa-large. We conducted over 300 experiments, consisting of approximately 3000 GPU hours using RTX6000 GPUs.

When implementing SLR-NLI with DeBERTa-base, hypotheses with more than 50 spans were not supervised (impacting only 0.02% training examples). This is due to memory constraints of the SLR-NLI method, which involves combining every span/premise pair into a single minibatch for each instance. This is memory intensive when hypotheses have a large number of spans and when premises are multiple sentences. For DeBERTa-large, we do not train with examples with more than 10 spans (impacting 11.46% of instances).

H Detailed Comparison of FactAI and FGLR

We find that most of the performance benefits from FGLR are from its ability to successfully distinguish between the contradiction and neutral classes.

	In-distribution				Out-of-distribution			
	R1	R2	R3	ANLI-all	ConTRoL	RTE	WNLI	Int?
DeBERTa-large	78.3	66.5	61.7	68.1	56.0	90.4	68.9	✗
Span atoms:								
SLR-NLI	74.7	60.4	58.3	64.1	54.7	87.5	65.8	✓
Sentence atoms:								
SenAI	75.3	63.7	59.1	65.6	53.4	86.1	64.7	✓
SENTLI	75.5	63.8	59.5	65.8	53.9	**86.4**	**65.4**	✓
SenLR (ours)	**76.7**‡	**64.8**‡	**62.0**‡	**67.5**‡	**56.3**‡	86.3	64.5	✓
Fact atoms:								
FactAI	70.0	60.2	57.3	62.2	48.3	81.0	**78.7**	✓
FGLR (ours)	**76.2** ‡	**64.8**‡	**63.1**‡	**67.7**‡	**52.7**‡	**82.0**†	77.0	✓

Table 2: Accuracy for DeBERTa-large. † represents results that are statistically better than the corresponding SenAI or FactAI baseline with $p < 0.05$, while ‡ represents results where $p < 0.01$, using bootstrapping statistical testing (Efron and Tibshirani, 1993). 'Int?' indicates whether the model is interpretable. All results displayed are an average from 10 different random seeds.

	In-distribution				Out-of-distribution			
	R1	R2	R3	ANLI-all	ConTRoL	RTE	WNLI	Int?
BERT-base	54.1	45.7	45.0	48.1	47.3	71.2	49.0	✗
Span atoms:								
SLR-NLI	51.5	42.5	42.7	45.4	46.5	73.9	44.4	✓
Sentence atoms:								
SenAI	56.0	46.1	**46.1**	**49.2**	44.5	69.1	52.3	✓
SENTLI	55.4	46.3	45.9	49.0	45.5†	69.8	**53.1**	✓
SenLR (ours)	**56.1**	**46.6**	45.7	**49.2**	**46.9**†	**71.5**‡	44.1	✓
Fact atoms:								
FactAI	55.3	44.5	44.8	48.0	44.0	65.4	**60.8**	✓
FGLR (ours)	**58.4**‡	**46.0**‡	**46.6**‡	**50.1**‡	**44.4**	**71.6**‡	55.8	✓

Table 3: Accuracy for BERT. [1] results are reported by (He et al., 2023). † represents results that are statistically better than the corresponding SenAI or FactAI baseline with $p < 0.05$, while ‡ represents results where $p < 0.01$, using bootstrapping statistical testing (Efron and Tibshirani, 1993). 'Int?' indicates whether the model is interpretable. All results displayed are an average from 10 different random seeds.

A qualitative analysis confirms that FGLR performs well in this respect, with FactAI often predicting contradiction when the information provided does not necessarily contradict the hypothesis. For example, knowing that 'Judy Tegart Dalton was a runner-up in 10... tournaments' does not contradict a hypothesis that she 'won more than nine... titles'. To provide empirical evidence of this finding, we try reducing the NLI task to deciding between 'entailment' and 'non-entailment' during inference, collapsing both the neutral and contradiction classes. In this case, we find that FactAI now outperforms FGLR on ANLI (with 73.7% accuracy compared to 72.3% for FGLR). This highlights how the performance advantages of FGLR are a result of its ability to successfully differentiate between the contradiction and neutral classes.

Our qualitative analysis also highlights that FGLR often predicts entailment when a fact most likely implies a premise, but when there is not full entailment. We do not see the same behaviour with the FactAI model, which is considerably less likely

	R1	R2	R3	ANLI-all
Ablation experiments:				
FGLR - No h-cond facts	69.6	54.4	52.4	58.4
FGLR - No instance loss	**71.8**	55.6	55.1	60.5
FGLR - No fact loss	42.2	39.2	37.4	39.5
Fact generation strategies:				
FGLR - Extended fact list	69.4	54.4	52.6	58.4
FGLR - Combined fact list	69.9	55.4	52.9	59.0
FGLR - Combined fact list & h-cond facts	70.9	**56.1**	55.0	60.3
FGLR - Extended fact list & h-cond facts	71.4	55.8	**55.6**	60.6
FGLR	**71.8**	**56.1**	55.3	**60.7**

Table 4: Ablation experiments (each an average from 10 seeds), comparing performance of our FGLR system to the following settings: 1) When the hypothesis-conditioned facts are not included (No h-cond facts) 2) the instance loss component is excluded from FGLR (No instance loss), and 3) when the fact loss component is excluded from FGLR (No fact loss). We also consider different fact generation strategies: 4) when the fact lists are extended, 5) when combining two independent fact lists, 6) when combining two independent fact lists with the hypothesis-conditioned facts, 7) when combining the extended fact list with the hypothesis-conditioned facts.

	R1	R2	R3	ANLI-all
DeBERTa-base baseline w/ facts appended	33.3	33.3	33.5	33.4
DeBERTa-base baseline	71.2	54.0	51.7	58.4
FGLR	**71.8**	**56.1**	**55.3**	**60.7**

Table 5: We experiment with appending the generated facts to the baseline model (baseline w/ facts appended), although the model does not converge in this setting. All results are an average of 10 seeds.

	R1	R2	R3	ANLI-all
FactAI	65.2	50.6	49.9	54.9
FactAI w/ h-cond facts	68.3	52.8	52.1	57.4
FGLR	**71.8**	**56.1**	**55.3**	**60.7**

Table 6: We experiment with providing the FactAI baseline with the hypothesis-conditioned facts, measuring the extent to which our fact-generation strategy can improve performance without training with atoms in-the-loop (FGLR). All results are an average of 10 seeds.

	R1	R2	R3	ANLI-all
FGLR w/ GPT-3.5-turbo (few-shot)	45.0	39.4	43.8	42.8
FGLR	**71.8**	**56.1**	**55.3**	**60.7**

Table 7: We experiment with replacing our BERT or DeBERTa models in FGLR with a GPT-3.5-turbo model (in a few-shot setting, with two examples provided in the prompt). This results in poor performance compared to FGLR. All results are an average of 10 seeds.

to predict the entailment class for an individual fact. To better understand the cause of this behaviour, we review the generated facts for 100 instances in the validation set, finding that in 21% of cases, there is no single fact that truly implies the entire hypoth-

	R1	R2	R3	ANLI-all
DeBERTa-base baseline	71.2	54.0	51.7	58.4
FGLR w/ no facts, just full NLI prem & hyp	**72.5**	54.7	52.2	59.3
FGLR	71.8	**56.1**	**55.3**	**60.7**

Table 8: We train our FGLR model using only the full premise and hypothesis, without including any fact-level or sentence-level decomposition. This results in only a small improvement compared to the DeBERTa-base baseline. All results are an average of 10 seeds.

	In-distribution	**Out-of-distribution**					
	ConTRoL	R1	R2	R3	ANLI-all	RTE	WNLI
DeBERTa-base	39.2	33.4	33.4	33.5	33.4	52.7	43.7
Fact atoms:							
FactAI	39.2	33.4	33.4	33.5	33.4	52.7	43.7
FGLR	**47.8**	**43.7**	**39.7**	**50.0**	**41.5**	**61.8**	**52.7**

Table 9: Training with ConTRoL, and testing on out-of-distribution NLI datasets (ANLI r1, r2, r3, RTE and WNLI). FGLR outperforms DeBERTa-base and FactAI, which both fail to converge in this setting. All results are an average of 10 seeds.

esis[8]. Sometimes this is caused by subtle reasons, for example, one hypothesis says 'Shostakovich may have been lying about his life in his book', while the relevant fact says 'Some consider the book Testimony to be a fabrication'. In this case, the fact is missing the information that Testimony is the name of Shostakovich's book. These findings suggest that further improvements to the generated facts are likely to further improve the performance of FGLR for entailment predictions.

We additionally analyse human annotations for each individual fact from the same 100 validation examples. To understand the respective strengths between FactAI and FGLR, we chose 2 seeds (out of 10) where both FactAI and FGLR have identical performance on these 100 examples (we also do not include the hypothesis-conditioned facts in this analysis). We find that FGLR has a lower F1 score for entailment, whereas FactAI has a lower F1 score for contradiction (see Table 11), supporting the findings from our qualitative analysis above.

We conclude that the performance improvements of FGLR are driven by its better performance on neutral and contradiction instances, which is reflected by better fact-level F1 performance on the contradiction class. On the other hand, we find that FactAI performs better on entailment examples, which is also reflected in the fact-level performance for the 100 annotated examples.

[8]This increases to 46% without the hypothesis-conditioned facts (which were not included during training)

Premise:

The Ottawa Sun is a daily tabloid newspaper in Ottawa, Ontario, Canada. It is published by Sun Media. It was first published in 1983 as the "Ottawa Sunday Herald", until it was acquired by (then) Toronto Sun Publishing Corporation in 1988. In April 2015, Sun Media papers were acquired by Postmedia.

Hypothesis:

Toronto Sun Publishing acquired the Ottawa Sun in the late nineties

Fact list:

Fact list, excluding hypothesis-conditioned fact

1. The Ottawa Sun is a daily tabloid newspaper in Ottawa, Ontario
2. The Ottawa Sun is a tabloid newspaper
3. The Ottawa Sun is published by Sun Media Toronto
4. The Ottawa Sun was first published in 1983 as the "Ottawa Sunday Herald"
5. The Ottawa Sun was acquired by (then) Toronto Sun Publishing Corporation in 1988
6. Sun Media papers were acquired by Postmedia in April 2015

Hypothesis-conditioned fact

The Ottawa Sun was acquired by Toronto Sun Publishing Corporation in 1988

Predictions:

#	FactAI	FGLR
1	Neutral	Neutral
2	Neutral	Neutral
3	Neutral	Neutral
4	Neutral	Neutral
5	Contradiction	Contradiction
6	Contradiction	Neutral
H-cond	N/A	Contradiction

*As both FactAI and FGLR have **at least one fact predicted as contradiction**, both models predict contradiction (the correct class). FactAI reaches the correct answer at an instance-level **despite not making a correct prediction for each individual fact**.*

Figure 3: We show the premise, hypothesis and the generated fact list for a hypothesis-premise pair in the dev set that both FactAI and FGLR correctly predict. However, despite correct instance-level predictions, we see FactAI predicting contradiction for fact #6, even when this is not appropriate. In this case, the acquisition in 2015 by Postmedia does not contradict there also being an acquisition by Toronto Sun Publishing in 1988. The hypothesis conditioned fact generated for FGLR is almost identical to fact #5, and FGLR predicts both facts as contradiction.

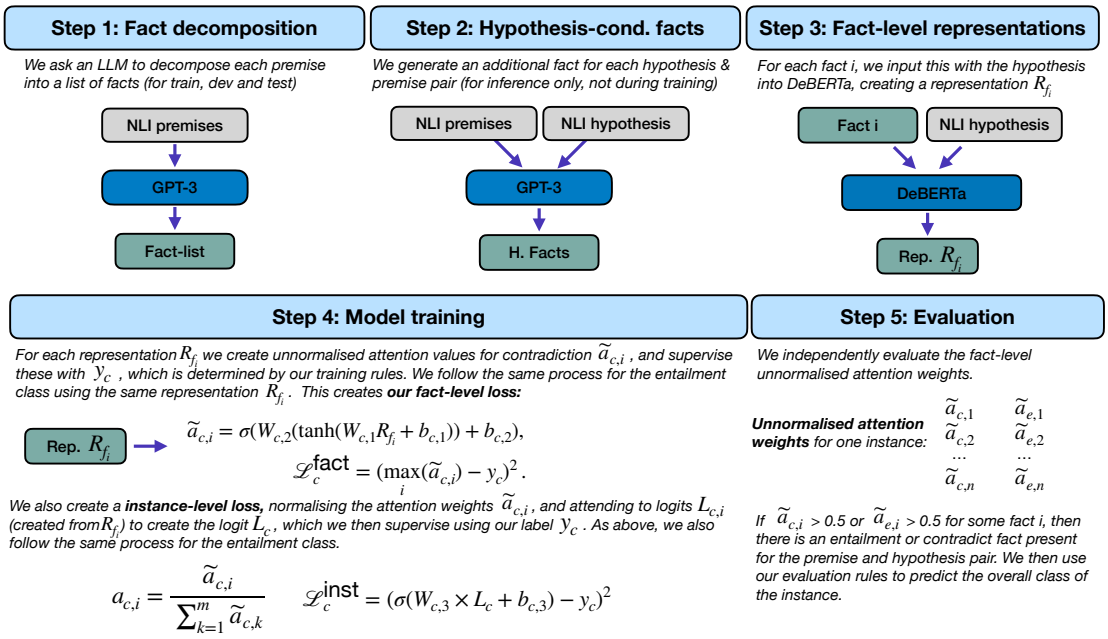

Figure 4: Our FGLR method is summarised above, with five stages: 1) generating fact lists for each premise, 2) generating an additional fact when performing inference, prompting GPT-3 to create a relevant fact from the premise for a specific hypothesis, 3) creating a representation for each fact, 4) our fact-level and instance-level losses used in training, and 5) evaluation using our evaluation rules.

Premise:

The Ottawa Sun is a daily tabloid newspaper in Ottawa, Ontario, Canada. It is published by Sun Media. It was first published in 1983 as the “Ottawa Sunday Herald”, until it was acquired by (then) Toronto Sun Publishing Corporation in 1988. In April 2015, Sun Media papers were acquired by Postmedia.

Hypothesis:

Toronto Sun Publishing acquired the Ottawa Sun in the late nineties

Fact list:

1. The Ottawa Sun is a daily tabloid newspaper in Ottawa, Ontario
2. The Ottawa Sun is a tabloid newspaper
3. The Ottawa Sun is published by Sun Media Toronto
4. The Ottawa Sun was first published in 1983 as the “Ottawa Sunday Herald”
5. The Ottawa Sun was acquired by (then) Toronto Sun Publishing Corporation in 1988
6. The Ottawa Sun was acquired by Toronto Sun Publishing Corporation in 1988
7. Sun Media papers were acquired by Postmedia in April 2015

Figure 5: The premise, hypothesis and the generated fact list for a hypothesis-premise pair in the dev set. The 6th fact is the hypothesis-conditioned fact (the content of this fact overlaps with the 5th fact provided). The example provided is from the DeBERTa-base FGLR model. FGLR correctly predicts facts 5 and 6 as contradiction (with all other facts being neutral).

Premise:

The Aberdeen Fortress Royal Engineers was a Scottish volunteer unit of the British Army formed in 1908. Its main role was defence of the Scottish coast, but it served on the Western Front during World War I. In the 1930s it was converted into an air defence unit, in which role it served in World War II.

Hypothesis:

The Aberdeen Fortress Royal Engineers served air defence in World War 1

Fact list:

1. The Aberdeen Fortress Royal Engineers was a Scottish volunteer unit of the British Army
2. The Aberdeen Fortress Royal Engineers was formed in 1908
3. The Aberdeen Fortress Royal Engineers was a unit of the British Army defence of the Scottish coast
4. The Aberdeen Fortress Royal Engineers served on the Western Front during World War I
5. The Aberdeen Fortress Royal Engineers was converted into an air defence unit in which role it served in World War II

Figure 6: The premise, hypothesis and the generated fact list for a hypothesis-premise pair in the dev set. The 4th fact is the hypothesis-conditioned fact. The example provided is from the DeBERTa-base FGLR model. FGLR correctly predicts the 5th fact as being a contradiction (with all other facts being neutral).

	In-distribution				Out-of-distribution			
	R1	R2	R3	ANLI-all	ConTRoL	RTE	WNLI	Int?
DeBERTa-base	1.05	1.25	0.93	0.66	1.51	1.90	2.82	✗
Span atoms:								
SLR-NLI	1.13	1.29	1.18	1.01	1.46	1.64	3.32	✓
Sentence atoms:								
SenAI	1.29	1.36	1.10	0.98	1.53	1.78	2.30	✓
SENTLI	1.33	1.21	1.10	0.97	1.03	1.86	2.15	✓
SenLR	0.62	0.96	0.88	0.60	1.67	1.34	2.47	✓
Fact atoms:								
FactAI	1.47	1.19	0.81	0.81	1.74	1.21	2.82	✓
FGLR	0.62	1.08	0.90	0.54	1.99	1.10	2.95	✓

Table 10: Standard deviations corresponding to the reported mean results in Table 1 after using 10 different random seeds.

	Precision	Recall	F1
FGLR no h-cond facts			
Entailment	0.24	0.76	0.36
Neutral	0.98	0.85	0.91
Contradiction	0.68	0.77	0.72
Macro average	0.63	0.79	0.66
FactAI			
Entailment	0.44	0.68	0.53
Neutral	0.97	0.91	0.94
Contradiction	0.55	0.76	0.64
Macro average	0.65	0.78	0.70

Table 11: We compare model fact-level predictions to human annotations for 100 examples in the validation set (using 2 random seeds)