Sentence Simplification for Natural Language to Formal Logic Translation

Roberto Milanese Jr.$^{1[0009-0009-5107-162X]}$, Adam Pease$^{1[0000-0001-9772-1266]}$, and Richard Thompson$^{1[0009-0001-6541-1092]}$

Naval Postgraduate School, Monterey, CA, USA
{roberto.milanese, adam.pease, richard.thompson}@nps.edu

Abstract. Translating natural language into formal logic enables automated reasoning, yet the syntactic and semantic complexity of natural text often prevents reliable formalization. This work presents a sentence simplification framework designed to improve language-to-logic translation within our language understanding pipeline. Our approach integrates large language models (LLMs) with in-context learning (ICL) and introduces an adversarial self-checking technique that iteratively detects and corrects errors. Results show that adversarial self-checking substantially improves output quality, enabling smaller models to outperform larger models. Across ICL strategies, curated static examples achieved the best overall performance, while syntactic retrieval outperformed semantic similarity. We further demonstrate that logic-tailored simplification, emphasizing clause splitting and pronoun resolution, improves syntactic validity and reduces hallucination rates in downstream logic translation. Together, these findings establish sentence simplification as a critical enabler for neuro-symbolic autoformalization, supporting more reliable and verifiable AI reasoning in analysis, policy, and decision-making contexts.

Keywords: Text Simplification · Language-to-Logic Translation · Neuro-Symbolic AI · In-Context Learning · Sentence Simplification · Automated Reasoning

1 Introduction

Natural language contains ambiguities, nested constructions, and pronouns that hinder translation into formal logic. This complexity prevents machines from reasoning reliably about human text, which limits the trustworthiness of artificial intelligence (AI) systems in high-stakes domains such as policy analysis or decision support. While AI models can generate fluent text, they are prone to hallucinations, contradictions, and errors that undermine confidence in their outputs [1, 2, 3].

The Hybrid Neuro-Ontological Language Understanding (HyNOLU) project [4] addresses these challenges by developing a pipeline for converting natural language into the formal language Standard Upper Ontology Knowledge Interchange Format (SUO-KIF) [5] using logical symbols from the Suggested Upper Merged Ontology (SUMO)[1] [6, 7]. Translating into an expressive logic enables formal reasoning with automated theorem provers such as Eprover [8] and Vampire [9]. YThe architecture combines preprocessing, language-to-logic (L2L) translation, and logical validation using automated theorem provers (ATPs). Within this pipeline, sentence simplification plays a critical role by restructuring complex language into atomic statements that are more tractable for downstream translation.

This paper focuses on the sentence simplification component of HyNOLU, which introduces two core contributions:

1. An adversarial self-checking loop in which models iteratively critique and revise simplifications until an acceptable output is produced.
2. An extension of this approach to a custom corpus designed for L2L simplification, targeting atomic decomposition and pronoun resolution for logic translation.

Experiments compare multiple large language models (LLMs) and in-context learning (ICL) strategies, with performance measured using established simplification metrics such as SARI and METEOR. Results show that adversarial self-checking consistently improves outcomes and enables smaller models to surpass larger ones. Static ICL examples selected for structural diversity yielded the strongest results, while syntactic similarity retrieval outperformed semantic similarity retrieval. Applying the framework to the L2L corpus produced inconclusive quantitative results due to the immaturity of downstream translation models, but qualitative assessment showed promising improvements in atomicity and pronoun handling.

2 Background

This section situates our work within prior research on sentence simplification and introduces the evaluation methods relevant to our experiments. We briefly review the evolution of simplification techniques, highlight commonly used datasets and metrics, and discuss recent advances in LLMs and ICL.

[1] `https://www.ontologyportal.org`

2.1 Sentence Simplification Approaches

Early efforts in text simplification were motivated by accessibility and parser performance, with hand-crafted rule systems targeting operations such as clause splitting, passive-to-active voice conversion, and lexical substitution [10, 11]. Statistical machine translation methods followed, modeling simplification as monolingual translation with operations including reordering, substitution, and deletion [12]. Subsequent work explored hybrid frameworks that combined syntactic transformations with statistical or semantic models [13].

With the advent of neural architectures, encoder–decoder models and reinforcement learning systems such as DRESS [14] achieved state-of-the-art results, optimizing directly for simplicity, fluency, and meaning preservation. More recently, controllable models such as ACCESS [15] introduced explicit simplification parameters, while LLMs have demonstrated strong zero- and few-shot capabilities across simplification tasks [16].

2.2 Datasets and Metrics

Simplification research has relied heavily on corpora aligned between complex and simple sentences. Early resources such as PWKP [12] and Newsela [17] enabled supervised training, though quality and availability were limiting factors. The TurkCorpus [18] introduced multiple human references per sentence, improving evaluation reliability. Building on this, the ASSET corpus [19] encouraged diverse edits across splitting, paraphrasing, and reordering, and has become a standard benchmark for modern systems. In this work, we also introduce a custom corpus tailored for L2L translation, designed to emphasize atomic decomposition and pronoun resolution.

Evaluation of simplification quality has traditionally relied on n-gram overlap metrics such as BLEU [20], ROUGE [21], and METEOR [22]. However, these metrics often fail to capture structural or semantic improvements. The SARI metric [18] addresses this gap by directly evaluating addition, deletion, and retention operations relative to the source and reference sentences. Our experiments therefore evaluate simplification quality using a full suite of metrics, including BLEU, ROUGE, METEOR, SARI, BERTScore [23], and FrugalScore [24].

2.3 Large Language Models and In-Context Learning

The emergence of LLMs such as GPT [25] and T5 [26] has transformed simplification research. These models can perform simplification without task-specific fine-tuning, guided instead by ICL. Recent work shows that even a handful of curated examples can match or surpass fine-tuned baselines [27, 16]. This motivates our systematic evaluation of ICL strategies, comparing static example sets, syntactic retrieval, and semantic retrieval.

2.4 Adversarial Self-Checking

Beyond initial generation, feedback mechanisms can improve output reliability. Recent studies in self-refinement and adversarial prompting demonstrate that iterative critique–revision cycles reduce hallucinations and improve factuality [28]. However, their application to text simplification has been limited. Our work adapts this paradigm by introducing an adversarial self-checking loop: models are prompted to critique their own simplifications, propose corrections if needed, and iterate until the output is acceptable.

2.5 Research Gap

Despite decades of progress, most simplification research focuses on human readability. Few works evaluate simplification explicitly for downstream reasoning or logic translation. Our contribution addresses this gap by systematically comparing ICL strategies and adversarial self-checking across multiple LLM backbones, and by extending the framework to an L2L corpus tailored for logic translation.

3 Methodology

We designed a sentence simplification pipeline aimed at improving downstream L2L translation within the HyNOLU framework. The pipeline was evaluated in two primary experiments: (1) benchmarking multiple ICL strategies against human-crafted simplifications in the ASSET corpus [19], and (2) assessing logic-tailored simplification on a custom corpus optimized for atomic decomposition and pronoun resolution.

3.1 ASSET Sentence Simplification Corpus

The ASSET corpus served as our primary benchmark, providing 2,000 validation and 359 test sentences, each paired with ten human-crafted simplifications. Multiple references increase evaluation reliability by capturing diverse simplification strategies. The validation set served as the pool of examples selected for ICL. To extend beyond human readability, we constructed a small custom corpus for L2L tasks, emphasizing clause splitting and pronoun resolution to reduce ambiguity in logic translation.

3.2 Model Selection

The models selected for testing represent a spectrum of size, architecture, and quantization strategy. Smaller models such as `mistral:7b-instruct-fp16` and `llama3.1:8b-instruct-q8_0` were included to evaluate performance in resource-constrained environments where memory and compute are limited. Larger models such as `llama3.3:70b-instruct-q4_K_M` and `phi4:14b-fp16` were chosen to test whether scaling parameter count and precision consistently yields improvements in simplification quality. The inclusion of `mistral-nemo:12b-instruct-2407-fp16` provides a middle ground between efficiency and capacity, highlighting trade-offs between runtime, resource usage, and output quality.

This diverse set enables a fair comparison across different scales of LLMs, helping to determine whether smaller, instruction-tuned models coupled with targeted prompting strategies can rival or even outperform larger, more resource-intensive alternatives.

3.3 In-Context Learning Strategies

Each input sentence was simplified by prepending demonstrations in the form `Original: <original sentence> /n Simplified: <simplified sentence>`, enabling the LLM to learn the transformation pattern. For each input, we selected the reference simplification with the greatest degree of sentence splitting, ensuring maximal decomposition for guiding the model. We tested four ICL strategies:

- **Static**: a fixed, hand-curated set of diverse simplifications.
- **Random**: randomly selected examples.
- **Semantic Retrieval**: nearest-neighbor search using Sentence-BERT embeddings [29].
- **Syntactic Retrieval**: retrieval based on tree edit distance between dependency parses [30].

The semantic method often retrieved sentences that were similar in meaning but structurally irrelevant, whereas syntactic retrieval favored structurally aligned but semantically unrelated sentences. Both dynamic strategies were compared against the static baseline. Figure 1 illustrates the retrieval process.

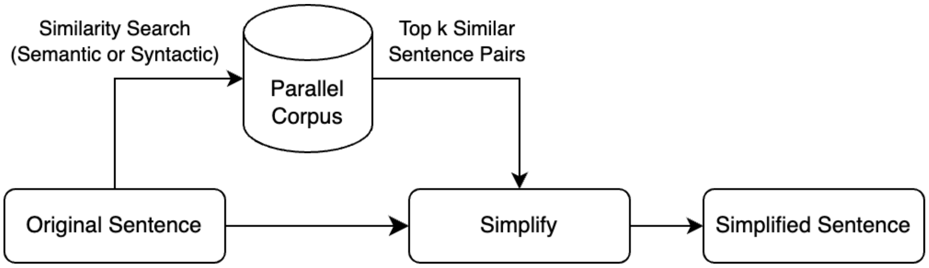

Fig. 1. ICL Retrieval Process

3.4 Adversarial Self-Checking Loop

To mitigate hallucinations and information loss, we implemented an adversarial self-checking loop, shown in Figure 2. After generating a simplification, a separate model instance critiqued it for three error types: added information, missing information, or unacceptable meaning change. The critique returned a JSON object with recommendations, which was fed back to the original model for revision. This process iterated until the critique accepted the output or a maximum retry threshold was reached. Adjusting the model temperature up to 0.5 improved convergence by preventing repetitive loops. Only `llama3.1:8b-instruct-q8_0` and `mistral:7b-instruct-fp16` were tested in this adversarial configuration, in order to evaluate whether smaller models could be improved for use in resource-constrained environments.

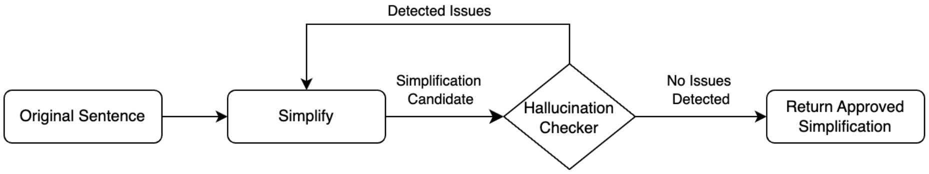

Fig. 2. Adversarial Simplification Loop

3.5 L2L Tailored Simplification

Human-oriented simplifications in the ASSET corpus often remain too complex for logic translation, particularly in cases with pronouns or coordinated noun phrases. To address this, we created handcrafted L2L examples that decompose sentences into atomic statements aligned with training data for logic models. For example:

> **Original**: The tornado destroyed 20 homes, left 30 others severely damaged, injured one person, and caused power outages.
>
> **ASSET Corpus Simplification**: The tornado destroyed 20 homes and damaged 30. It injured one person, and caused power outages.
>
> **L2L Simplification**: The tornado destroyed 20 homes. The tornado damaged 30 homes. The tornado injured one person. The tornado caused a power outage for 30 people.

We further applied pronoun resolution to ensure referential grounding before SUO-KIF translation, using a lightweight LLM-based resolver.

3.6 Evaluation Metrics

Generated simplifications were evaluated against ASSET corpus references using the following metrics:

- system output against references and against the input sentence (SARI) [18]

- metric for evaluation of translation with explicit ordering (METEOR) [22]
- BLEU [20]
- ROUGE [21]
- BERTScore
- FrugalScore

For L2L evaluation, simplified and original sentences were translated into SUO-KIF using fine-tuned T5 and LLaMA models, and outputs were assessed for syntax validity, hallucinated terms, and aggregate validity scores.

4 Results

We evaluated 359 test sentences from the ASSET corpus using multiple LLMs and ICL configurations. Simplification quality was measured with BLEU, METEOR, ROUGE-L, SARI, BERTScore, and FrugalScore. We then applied a logic-tailored pipeline on 105 complex sentences to test downstream L2L translation effects.

4.1 Simplification Performance

Top Configurations Of the 67 unique configurations tested, the adversarial self-checking pipeline paired with `llama3.1:8b` achieved the highest overall performance, surpassing larger models such as `phi4:14b` and `llama3.3`. The top ten scoring configuration are displayed in Table 1.

Table 1. Top-performing configurations with associated average simplification metrics.

Model	**ICL Type / Size**	**SARI**	**FrugalScore**	**BERTScore**	**ROUGE-L**	**METEOR**	**BLEU**
llama3.1:8b_adv	dynamic_tree / 5	0.47	**0.60**	**0.98**	**0.84**	**0.89**	**0.76**
phi4:14b-fp16	static / 5	**0.48**	0.59	0.97	0.81	0.87	0.72
llama3.3	static / 5	**0.48**	0.59	0.97	0.81	0.86	0.73
llama3.3	static / 10	**0.48**	0.58	0.97	0.80	0.86	0.72
phi4:14b-fp16	static / 10	**0.48**	0.59	0.97	0.80	0.86	0.71
llama3.1:8b	static / 5	0.46	0.58	0.97	0.81	0.86	0.73
llama3.1:8b	dynamic_tree / 1	0.45	0.57	0.97	0.81	0.83	0.75
llama3.1:8b	static / 1	0.46	0.58	0.97	0.80	0.86	0.72
phi4:14b-fp16	dynamic_tree / 5	0.46	0.57	0.97	0.80	0.84	0.71
llama3.1:8b	static / 10	0.46	0.58	0.97	0.79	0.84	0.70

Model-Level Comparison Averaging across all ICL strategies, smaller quantized models outperformed larger ones on multiple metrics. The results are displayed in Table 2. The `llama3.1:8b` model achieved the highest BLEU and ROUGE-L, while `phi4:14b` led in SARI and METEOR.

Table 2. Average simplification performance by model.

Model	SARI	Frugal	BERT	ROUGE-L	METEOR	BLEU
llama3.1:8b	0.44	0.56	**0.97**	**0.79**	0.82	**0.71**
phi4:14b	**0.46**	**0.57**	**0.97**	0.78	**0.83**	0.67
llama3.3	0.45	0.54	**0.97**	0.77	0.81	0.68
mistral-nemo	0.42	0.55	**0.97**	0.76	0.78	0.65
mistral:7b	0.41	0.51	**0.97**	0.74	0.75	0.61

ICL Strategy and Size Static examples with handpicked diversity yielded the strongest results (Table 3). Syntactic retrieval also performed well, while semantic retrieval offered no advantage over random selection.

Table 3. Average performance by ICL type.

ICL Type	SARI	Frugal	BERT	ROUGE-L	METEOR	BLEU
static	**0.45**	**0.56**	**0.97**	**0.78**	**0.82**	**0.68**
syntactic_tree	0.44	0.55	**0.97**	**0.78**	0.80	**0.68**
random	0.43	0.54	**0.97**	0.77	0.79	0.66
semantic_sim	0.44	0.55	**0.97**	0.76	0.79	0.65
none	0.40	0.51	0.96	0.72	0.74	0.59

As shown in Table 4, a single in-context example provided major improvements over zero-shot. Beyond five examples, gains plateaued.

Table 4. Average performance by ICL size.

Size	SARI	Frugal	BERT	ROUGE-L	METEOR	BLEU
10	**0.45**	**0.56**	**0.97**	**0.78**	**0.81**	0.67
5	0.44	**0.56**	**0.97**	**0.78**	**0.81**	0.67
1	0.43	0.54	**0.97**	0.77	0.79	0.66
0	0.40	0.51	0.96	0.72	0.74	0.59

Adversarial Self-Checking The adversarial loop (Table 5) yielded the strongest results overall. Crucially, it elevated smaller models above larger baselines, demonstrating that structured feedback can substitute for scale.

Table 5. Adversarial self-checking results (syntactic_tree / size 5).

Model	SARI	Frugal	BERT	ROUGE-L	METEOR	BLEU
llama3.1:8b_adv	**0.47**	**0.60**	**0.98**	**0.84**	**0.89**	**0.76**
mistral:7b_adv	0.45	0.56	0.97	0.79	0.83	0.71

4.2 Logic-Tailored Simplification

We observed that logic-tailored simplification yielded more atomic and pronoun-resolved structures. For example:

> **Original Sentence:** Saint Martin is a tropical island in the northeast Caribbean, approximately 300 km (186 miles) east of Puerto Rico.
> **Original Simplification:** Saint Martin is a tropical island in the northeast Caribbean. It is approximately 300 km east of Puerto Rico.
> **L2L-Tailored:**
> - Saint Martin is a tropical island.
> - Saint Martin is in the northeast Caribbean.
> - Saint Martin is approximately 300 km (186 miles) east of Puerto Rico.

Each L2L-Tailored sentences contain minimal numbers of facts, facilitating simpler logic statements needed to formally represent sentences. Further research is needed to quantitatively assess this impact. Table 6 shows slight improvements in syntax validity of natural language sentences translated into formal SUO-KIF.

Table 6. SUO-KIF results before vs. after logic-tailored simplification.

Model/Input	Validity %	Avg. Terms
LitGPT Orig	94.3	15.0
LitGPT Simpl	95.2	17.9
T5-FLAN Orig	93.3	15.4
T5-FLAN Simpl	95.0	19.1

5 Discussion

The results highlight three main findings.

5.1 ICL Strategies

Static examples outperformed other strategies, underscoring the value of curated demonstrations with structural diversity. Syntactic retrieval proved more useful than semantic similarity, confirming that structural alignment matters more than meaning for simplification. Zero-shot was weakest, showing that even minimal in-context guidance substantially improves performance.

5.2 Adversarial Loop

The self-checking loop produced the largest gains, especially for smaller models. By iteratively identifying hallucinations and meaning shifts, models refined their own outputs without external supervision. This allowed `llama3.1:8b` and `mistral:7b` to outperform much larger models, suggesting that feedback mechanisms can substitute for scale in resource-constrained environments.

5.3 Logic-Tailored Simplification

Quantitative SUO-KIF improvements were modest, but qualitative inspection showed clearer agent–patient roles in generated SUO-KIF, better pronoun resolution in simplified sentences, and more atomic decomposition. These traits align closely with the needs of symbolic translation, even if current downstream L2L models lack the strength to fully capitalize on the improvements.

5.4 Limitations

There are several limitations with the adversarial approach. First, automatic metrics penalize simplifications that diverge lexically from references, even if they are subjectively better. Second, the ASSET corpus contains noisy or malformed sentences, limiting reliability. Third, logical fluency remains challenging, since over-splitting sentences risks losing temporal or causal relations. Future work should explore better L2L evaluators, curated corpora, and adaptive simplification strategies.

6 Conclusion

This work introduced a sentence simplification framework for Language-to-Logic (L2L) tasks. Through extensive benchmarking of multiple LLMs and ICL strategies using both traditional and modern evaluation metrics, we demonstrated the value of careful model and prompt design. An adversarial self-checking loop further enabled smaller models to outperform larger ones, while experiments showed that syntactic alignment in ICL selection is more effective than semantic similarity. In addition, logic-tailored simplification yielded qualitative improvements in SUO-KIF generation, highlighting the benefits of domain-specific adaptation.

Future work should focus on evaluation methods that integrate human judgment with automated scoring, the construction of larger and cleaner simplification corpora beyond ASSET, and the development of stronger L2L models capable of leveraging atomic simplifications. By combining syntactic retrieval for ICL with adversarial self-checking, this research advances the creation of explainable and verifiable reasoning pipelines. Although challenges remain, the proposed framework provides a concrete step toward trustworthy L2L translation.

References

[1] Yue Zhang et al. "Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models". In: *Comput. Linguist.* (2025). DOI: 10.1162/coli.a.16.

[2] Parshin Shojaee et al. *The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity.* arXiv. 2025. DOI: 10.48550/arXiv.2506.06941. eprint: 2506.06941 (cs.AI).

[3] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. *Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models.* arXiv. 2017. DOI: 10.48550/arXiv.1708.08296. eprint: 1708.08296 (cs.AI).

[4] Richard Thompson et al. "Formalizing Natural Language: Cultivating LLM Translations Using Automated Theorem Proving". In: *Theorem Proving and Machine Learning in the Age of LLMs: State of the Art and Future Perspectives.* EuroProofNet, COST Action CA20111 – European Research Network on Formal Proofs. Edinburgh, Scotland, UK, Apr. 2025. URL: https://europroofnet.github.io/wg5-edinburgh25/.

[5] Adam Pease. *SUO-KIF Reference Manual.* https://github.com/ontologyportal/sigmakee/blob/master/suo-kif.pdf. retrieved 20 June 2025. 2009.

[6] Ian Niles and Adam Pease. "Toward a Standard Upper Ontology". In: *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001).* Ed. by Chris Welty and Barry Smith. 2001, pp. 2–9.

[7] Adam Pease. *Ontology: A Practical Guide.* Angwin, CA: Articulate Software Press, 2011.

[8] Stephan Schulz. "E – a brainiac theorem prover". In: *AI Commun.* 15.2-3 (2002), pp. 111–126. DOI: 10.3233/EAI-2002-260.

[9] Laura Kovács and Andrei Voronkov. "First-order theorem proving and Vampire". In: *Int. Conf. on Comput. Aided Verif.* 2013. DOI: 10.1007/978-3-642-39799-8_1.

[10] Raman Chandrasekar, Christine Doran, and Srinivas Bangalore. "Motivations and Methods for Text Simplification". In: *16th Int. Conf. on Comput. Linguistics.* 1996. URL: https://aclanthology.org/C96-2183.pdf.

[11] John Carroll et al. "Practical Simplification of English Newspaper Text to Assist Aphasic Readers". In: *Proc. AAAI-98 Workshop on Integr. Artif. Intell. Assist. Technol.* 1998, pp. 7–10.

[12] Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. "A Monolingual Tree-based Translation Model for Sentence Simplification". In: *Proc. 23rd Int. Conf. on Comput. Linguist. (COLING 2010).* Aug. 2010, pp. 1353–1361. URL: https://aclanthology.org/C10-1152.

[13] Shashi Narayan and Claire Gardent. "Hybrid Simplification using Deep Semantics and Machine Translation". In: *Proc. 52nd Annu. Meet. Assoc.*

Comput. Linguistics (Vol. 1: Long Papers). ACL 2014. 2014, pp. 435–445. URL: https://aclanthology.org/P14-1041.

[14] Xingxing Zhang and Mirella Lapata. *Sentence Simplification with Deep Reinforcement Learning.* arXiv. July 16, 2017. DOI: 10.48550/arXiv.1703.10931.

[15] Louis Martin et al. *Controllable Sentence Simplification.* arXiv. 2020. DOI: 10.48550/arXiv.1910.02677. eprint: 1910.02677.

[16] Subha Vadlamannati and Gözde Gül Şahin. *Metric-Based In-context Learning: A Case Study in Text Simplification.* arXiv. 2023. DOI: 10.48550/arXiv.2307.14632. eprint: 2307.14632 (cs.CL).

[17] Chao Jiang et al. *Neural CRF Model for Sentence Alignment in Text Simplification.* 2021. DOI: 10.48550/arXiv.2005.02324. arXiv: 2005.02324 [cs.CL].

[18] Wei Xu et al. "Optimizing Statistical Machine Translation for Text Simplification". In: *Trans. Assoc. Comput. Linguist.* 4 (2016), pp. 401–415. DOI: 10.1162/tacl_a_00107.

[19] Fernando Alva-Manchego et al. *ASSET: A Dataset for Tuning and Evaluation of Sentence Simplification Models with Multiple Rewriting Transformations.* arXiv. 2020. DOI: 10.48550/arXiv.2005.00481.

[20] Kishore Papineni et al. "BLEU: A method for automatic evaluation of machine translation". In: *Proc. 40th Annu. Meet. Assoc. Comput. Linguistics.* 2002. DOI: 10.3115/1073083.1073135.

[21] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". In: *Text Summarization Branches Out.* 2004. URL: https://aclanthology.org/volumes/W04-10/.

[22] Satanjeev Banerjee and Alon Lavie. "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments". In: *Proc. ACL Workshop on Intrinsic and Extrinsic Eval. Measures for Mach. Transl. and/or Summarization.* 2005, pp. 65–72. URL: https://aclanthology.org/W05-0909.pdf.

[23] Tianyi Zhang et al. "BERTScore: Evaluating Text Generation with BERT". In: *Proc. 8th Int. Conf. on Learn. Represent.* 2020. DOI: 10.48550/arXiv.1904.09675.

[24] Wei Zhao et al. *MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover Distance.* arXiv. 2020. DOI: 10.48550/arXiv.1909.02622.

[25] Tom B. Brown et al. *Language Models are Few-Shot Learners.* arXiv. 2020. DOI: 10.48550/arXiv.2005.14165. eprint: 2005.14165 (cs.CL).

[26] Colin Raffel et al. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.* 2023. DOI: 10.48550/arXiv.1910.10683. arXiv: 1910.10683 [cs.LG].

[27] Jiachang Liu et al. *What Makes Good In-Context Examples for GPT-3?* arXiv. 2021. DOI: 10.18653/v1/2022.deelio-1.10. eprint: 2101.06804.

[28] Aman Madaan et al. *Self-Refine: Iterative Refinement with Self-Feedback.* 2023. arXiv: 2303.17651 [cs.CL]. URL: https://arxiv.org/abs/2303.17651.

[29] Nils Reimers and Iryna Gurevych. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.* arXiv. 2019. DOI: 10.48550/arXiv.1908.10084.

[30] Kaizhong Zhang and Dennis Shasha. "Simple Fast Algorithms for the Editing Distance between Trees and Related Problems". In: *SIAM J. Comput.* 18.6 (1989), pp. 1245–1262. DOI: 10.1137/0218082.