Project Proposal: Autoformalization for Algebras *

Bernardo Atalaia[1] and Mikoláš Janota[2] and João Araújo[1]

[1] Universidade Nova de Lisboa, Lisbon, Portugal
[2] Czech Technical University in Prague, Prague, Czechia

1 Introduction

Autoformalization is the task of automatically translating mathematical content written in natural language into formal language expressions. This process has gained significant traction due to advancements in the language interpretation capabilities of Large Language Models (LLMs). These models, with their expanding proficiency in formal languages, are progressively lowering the barriers to effective autoformalization. However, LLMs alone are insufficient to consistently and reliably deliver autoformalization, especially as the complexity and specialization of the target domain increase.

A major obstacle for autoformalization is the lack of training data. Therefore, often, data-generation for this task is needed [3].

This project targets data generation for the autoformalization task in the context of universal algebra.

2 Project Plan

In this project, we present a framework that leverages the Phi4 (14B) model [5, 6] to informalize mathematical expressions from the MARCIE [2] database, which contains around 400 formal classes represented in Prover9 [7]. In addition to informalizing these classes, the framework also generates new equivalent representations by employing a tool called Replace, part of ProverX [1] (A tool developed to integrate Prover9 with a scripting language), which systematically alters axioms while preserving the rules and semantics of the class involved. This process generates a diverse set of equivalent classes, which differ in this case in one axiom only, ensuring that the generated class with the new axiom is still equivalent to the original class. With this process, we can produce a significant amount of new classes generating tons of equivalent representations for each axiom of all classes. This is particularly useful by ensuring quality of the data aswell as size, a crucial factor for using/training LLMs.

To ensure the validity of the informalization process, our framework employs a verification method that formalizes the generated informal expressions back into Prover9 syntax and checks their equivalence with the original expressions. This validation guarantees a high-quality informalization when the equivalency holds. By providing a reliable and scalable approach to informalization, this work significantly contributes to advancing autoformalization tasks, particularly in the context of creating robust mathematical databases, to further use with the proverX tool [1].

*Supported by the Czech MEYS under the ERC CZ project no. LL1902 *POSTMAN*, by the European Union under the project *ROBOPROX* (reg. no. CZ.02.01.01/00/22_008/0004590).

3 Background and Related Work

Recent advances in machine learning have sparked interest in autoformalization [8, 10]. Welleck et al. [9] demonstrated the potential of LLMs in bridging natural and formal languages for mathematical reasoning. Similarly, Jiang et al. [4] explored how LLMs can assist in drafting formal proofs.

Paster et al. [3] introduced MathConstruct, a framework for generating synthetic data to support mathematical formalization tasks. Our approach builds upon this concept but focuses specifically on algebraic structures.

4 Methodology

Our methodology consists of three main phases:

1. **Data Collection:** We utilize the MARCIE database [2] as our primary source of formalized algebraic structures, generating even more examples with the replace function.

2. **Informalization:** We employ the Microsoft Phi-4 model [5, 6] to translate formal expressions into natural language while preserving the mathematical meaning.

3. **Validation:** We verify the quality of informalization by converting the natural language back to formal language and checking equivalence with the original formalization using ProverX [1].

5 Expected Outcomes

We anticipate this project will yield:

- A comprehensive dataset of paired formal-informal representations of algebraic structures
- Enhanced techniques for informalization and formalization of algebraic expressions
- Insights into how LLMs comprehend and manipulate algebraic concepts
- A robust framework for extending autoformalization to other mathematical domains

References

[1] ProverX, 2025. URL: https://www.proverx.dm.fct.unl.pt/doc/index.html.

[2] João Araújo and João Ramires. MarcieX: A model/theory/theorem database explorer. https://marciedb.pythonanywhere.com/, 2022. Accessed: 2025-05-16.

[3] Jasper Dekoninck, Mislav Balunovic, Nikola Jovanović, Ivo Petrov, and Martin Vechev. MathConstruct: Challenging LLM reasoning with constructive proofs. In *ICLR 2025 Workshop: VerifAI: AI Verification in the Wild*, 2025. URL: https://openreview.net/forum?id=nHW2tiGMrb.

[4] Albert Q. Jiang, Sean Welleck, Jin Peng Zhou, Wenda Li, Jiacheng Liu, Mateja Jamnik, Timothée Lacroix, Yuhuai Wu, and Guillaume Lample. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. *arXiv preprint arXiv:2210.12283*, 2022. URL: https://arxiv.org/abs/2210.12283.

[5] Ece Kamar. Introducing Phi-4: Microsoft's newest small language model specializing in complex reasoning, December 2024. URL: https://techcommunity.microsoft.com/blog/aiplatformblog/introducing-phi-4-microsoft%E2%80%99s-newest-small-language-model-specializing-in-comple/4357090.

[6] Kinfey Lo. Welcome to the new Phi-4 models - Microsoft Phi-4-mini & Phi-4-multimodal , February 2025. URL: https://techcommunity.microsoft.com/blog/educatordeveloperblog/welcome-to-the-new-phi-4-models---microsoft-phi-4-mini--phi-4-multimodal/4386037.

[7] William McCune. Prover9 and Mace4. *Department of Mathematics, The University of New Mexico*, 2005. URL: https://www.cs.unm.edu/~mccune/prover9/.

[8] Christian Szegedy. A promising path towards autoformalization and general artificial intelligence. In *International Conference on Intelligent Computer Mathematics*, volume 12236 of *Lecture Notes in Computer Science*, pages 3–20. Springer, 2020. doi:10.1007/978-3-030-53518-6_1.

[9] Sean Welleck, Jiacheng Liu, Ximing Lu, Hannaneh Hajishirzi, and Yejin Choi. Naturalprover: Grounded mathematical proof generation with language models. *arXiv preprint arXiv:2205.12910*, 2022. URL: https://arxiv.org/abs/2205.12910.

[10] Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. Autoformalization with large language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:29577–29589, 2022. URL: https://arxiv.org/abs/2205.12615.