ShZZaM: An LLM+ATP Natural Language to Logic Translator

First1 Last1, First2 Last2
Affiliation
City, Country

Abstract

This paper describes the ShZZaM tool that uses Large Language Models (LLMs) and Automated Theorem Proving (ATP) tools to translate natural language to typed first-order logic in the TFF syntax of the TPTP World.

Large Language Models (LLMs) (Peykani et al. 2025) have shown themselves to be useful in a broad range of applications (Sajjadi Mohammadabadi et al. 2025). However, it is well known that LLMs make mistakes (Huang et al. 2025), and this is acknowledged on LLMs' web interfaces, e.g., ChatGPT admits "ChatGPT can make mistakes. Check important info". In the face of such unreliability, the results from LLMs in mission-critical applications require verification. One approach is to translate the LLM input and output to a logical form that can be checked using Automated Theorem Proving (ATP) tools, e.g., (Yang et al. 2025; Cheng et al. 2025).[1] A key step in this verification pipeline is the faithful translation of the natural language to an appropriate logical form. This task is difficult due to the ambiguous nature of natural language statements, especially informally expressed statements. Work in this area includes LINC (Olausson et al. 2023), FOLIO (Han et al. 2024), and LINA (Li et al. 2024). This paper makes another contribution in this area, taking a new interactive approach to the translation process, zigzagging (hence the 'ZZ' in the tool name) between natural language and logic until convergence is achieved. A key feature of ShZZaM is its use of LLMs and Automated Theorem Proving (ATP) tools, which complement each other in the translation steps.

Figure 1 shows the overall process implemented of ShZZaM. Starting with the natural language, a combination of LLMs and ATP tools make a first translation (step 1 - a "Zig") to the typed first-order logic in the TFF syntax (Sutcliffe et al. 2012; Blanchette and Paskevich 2013) of the TPTP World (Sutcliffe 2024). An LLM is then used to translate the logic back to natural language (step 2 - a "Zag"). An LLM is then used to judge (step 3) whether or not the new and previous natural language statements have the same meaning. If they are very dissimilar - below the "acceptance threshold" - the Zig step is rejected, and is repeated. If they are adequately similar - above the "convergence threshold", the logic inbetween them is accepted as the translation. If the similarity lies between the acceptance and convergence thresholds then another zigzag is performed. This zizagging continues until the natural language pairs converge to the required level of similarity (or a limit is reached). Upon convergence the logic is sent to an ATP system via the SystemOnTPTP service (Sutcliffe 2000), either a model finder if there are only axioms in the logic, or a theorem prover if there is also a conjecture. The results from the ATP system is reported in the SZS format (Sutcliffe 2008).

Step 1, the translation from natural language to TFF logic, is an iterative one involving LLMs and ATP tools. Figure 2 shows the details. An LLM is used to translate from natural language to logic. The translation is successively checked for syntax errors, and if successful for type errors (recall the logic is *typed* first-order logic). If an error occurs in either check the error message is captured, and passed back into the LLM for another attempt. Note that after the logic passes the syntax and type checks, there is another outer level of looping based on the similarity of the previous and new natural language, as explained above.

The LLM translation from natural language to logic and back again works because the LLM has been exposed to enough natural language and enough TPTP format TFF logic. The former is simply the results of scraping the world's web sites, etc. The latter might be surprising, as TFF is comnparatively speaking a small fragment of the data used to train the LLM. Evidently there are adequate corpora that use TFF that are exposed on the web, e.g., the TPTP problem library (Sutcliffe 2017), exports from the Isabelle Archive of Formal Proofs (Blanchette et al. 2015), exports of the Mizar Mathematical Library (Urban 2003), etc.

Evaluation

This paper describes the derivation and interpretation viewers in the TPTP World: the Interactive Derivation Viewer (IDV), the Interactive Tableau Viewer (ITV), the Interactive Interpretation Viewer (IIV), and the Interactive Kripke Viewer (IKV). Users and developers of ATP systems are able to examine their ATP solutions in an interac-

[1] For a more comprehensive survey, just ask your favourite LLM to "show me some research on how LLMs make mistakes, and the need for symbolic checking of LLM output".

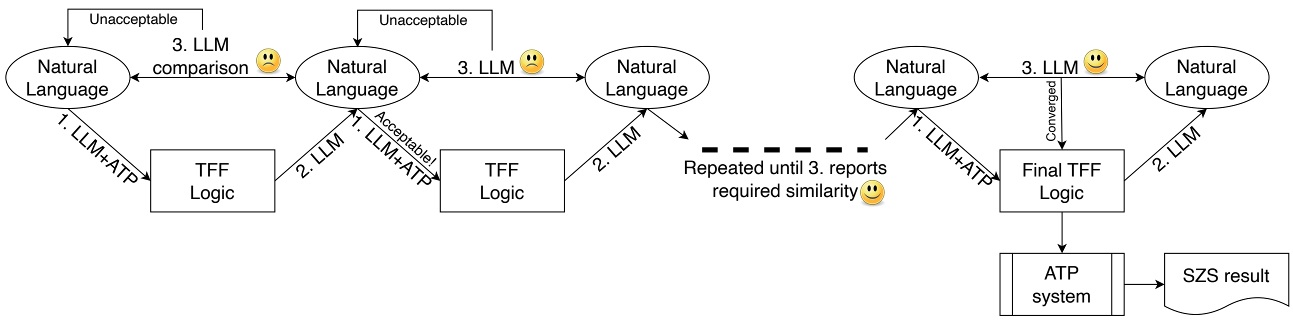

Figure 1: ShZZaM process

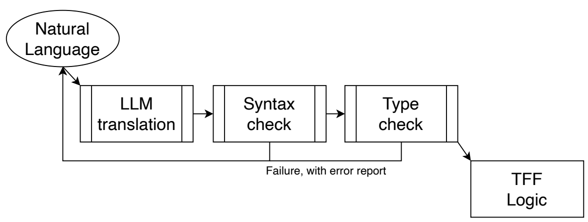

Figure 2: Translation from natural language to TFF logic

tive graphical environment, providing insights into features of the solutions. The viewers are freely accessible through SystemOnTPTP.

References

Blanchette, J., and Paskevich, A. 2013. TFF1: The TPTP Typed First-order Form with Rank-1 Polymorphism. In Bonacina, M., ed., *Proceedings of the 24th International Conference on Automated Deduction*, number 7898 in Lecture Notes in Artificial Intelligence, 414–420. Springer-Verlag.

Blanchette, J.; Haslbeck, M.; Matichuk, D.; and Nipkow, T. 2015. Mining the Archive of Formal Proofs. In Kerber, M.; Carette, J.; Kaliszyk, C.; Rabe, F.; and Sorge, V., eds., *Proceedings of the 8th Conference on Intelligent Computer Mathematics*, number 9150 in Lecture Notes in Computer Science, 3–17. Springer-Verlag.

Cheng, F.; Li, H.; Liu, F.; Van Rooij, R.; Zhang, K.; and Lin, Z. 2025. Empowering LLMs with Logical Reasoning: A Comprehensive Survey. In Kwok, J., ed., *Proceedings of the 34th International Joint Conference on Artificial Intelligence*, 10400–10408. AAAI Press.

Han, S.; Schoelkopf, H.; Zhao, Y.; Qi, Z.; Riddell, M.; Zhou, W.; Coady, J.; Peng, D.; Qiao, Y.; Benson, L.; Sun, L.; Wardle-Solano, A. Szabó, H.; Zubova, E.; Burtell, M.; Fan, J.; Liu, Y.; Wong, B.; Sailor, M.; Ni, A.; Nan, L.; Kasai, J.; Yu, T.; Zhang, R.; Fabbri, A.; Kryscinski, W.; Yavuz, S.; Liu, Y.; Lin, X.; Joty, S.; Zhou, Y.; Xiong, C.; Ying, R.; Cohan, A.; and Radev, D. 2024. FOLIO: Natural Language Reasoning with First-Order Logic. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 22017–22031. Association for Computational Linguistics.

Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; and Liu, T. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems* 42:1–55.

Li, Q.; Li, J.; Liu, T.; Zeng, Y.; Cheng, M.; Huang, W.; and Liu, Q. 2024. Leveraging LLMs for Hypothetical Deduction in Logical Inference: A Neuro-Symbolic Approach.

Olausson, T.; Gu, A.; Lipkin, B.; Zhang, C.; Solar-Lezama, A.; Tenenbaum, J.; and Levy, R. 2023. LINC: A Neurosymbolic Approach for Logical Reasoning by Combining Languag Models with First-Order Logic Provers. In Pino, J., and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 5153–5176. Association for Computational Linguistics.

Peykani, P.; Ramezanlou, F.; Tanasescu, C.; and Ghanidel, S. 2025. Large Language Models: A Structured Taxonomy and Review of Challenges, Limitations, Solutions, and Future Directions. *Applied Sciences* 15(14):8103.

Sajjadi Mohammadabadi, S.; Kara, B.; Eyupoglu, C.; Uzay, C.; Tosun, M.; and Karakuş, O. 2025. A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications. *Electronics* 14(18):3580.

Sutcliffe, G.; Schulz, S.; Claessen, K.; and Baumgartner, P. 2012. The TPTP Typed First-order Form with Arithmetic. In Bjørner, N., and Voronkov, A., eds., *Proceedings of the 18th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, number 7180 in Lecture Notes in Artificial Intelligence, 406–419. Springer-Verlag.

Sutcliffe, G. 2000. SystemOnTPTP. In McAllester, D., ed., *Proceedings of the 17th International Conference on Automated Deduction*, number 1831 in Lecture Notes in Artificial Intelligence, 406–410. Springer-Verlag.

Sutcliffe, G. 2008. The SZS Ontologies for Automated Reasoning Software. In Sutcliffe, G.; Rudnicki, P.; Schmidt, R.; Konev, B.; and Schulz, S., eds., *Proceedings of the LPAR Workshops: Knowledge Exchange: Automated Provers and Proof Assistants, and the 7th International Workshop on the*

Implementation of Logics, number 418 in CEUR Workshop Proceedings, 38–49.

Sutcliffe, G. 2017. The TPTP Problem Library and Associated Infrastructure. From CNF to TH0, TPTP v6.4.0. *Journal of Automated Reasoning* 59(4):483–502.

Sutcliffe, G. 2024. Stepping Stones in the TPTP World. In Benzmüller, C.; Heule, M.; and Schmidt, R., eds., *Proceedings of the 12th International Joint Conference on Automated Reasoning*, number 14739 in Lecture Notes in Artificial Intelligence, 30–50.

Urban, J. 2003. Translating Mizar for First Order Theorem Provers. In Asperti, A.; Buchberger, B.; and Davenport, J., eds., *Proceedings of the 2nd International Conference on Mathematical Knowledge Management*, number 2594 in Lecture Notes in Computer Science, 203–215. Springer-Verlag.

Yang, X.-W.; J-J., S.; Guo, L.-Z.; Zhang, B.-W.; Zhou, Z.; Jia, L.-H.; Dai, W.-Z.; and Li, Y.-F. 2025. Neuro-Symbolic Artificial Intelligence: Towards Improving the Reasoning Abilities of Large Language Models. In Kwok, J., ed., *Proceedings of the 34th International Joint Conference on Artificial Intelligence*, 10770–10778. AAAI Press.