Multi-MALLS: A Multilingual Benchmark for Natural Language to First-Order Logic Translation

Anonymous ACL submission

Abstract

Translating natural language to First-Order Logic (NL2FOL) is crucial for logic-based reasoning, but most existing models and benchmarks focus on English, leaving cross-lingual generalization underexplored. In this study, we present the first comprehensive investigation of cross-lingual robustness in NL2FOL. To support this, we introduce Multi-MALLS, the first multilingual benchmark for NL2FOL, which extends the widely-used MALLS dataset with high-quality translations into multiple languages. Using Multi-MALLS, we observe that state-of-the-art models experience a significant performance drop when applied to non-English inputs, highlighting the lack of robustness. To address this, we propose MA-FOL, a multi-agent framework that improves generalization without requiring any training on target languages. By decomposing the task into three modules, a language-agnostic structure generator, a language-specific predicate combiner, and a refinement component, MA-FOL achieves robust zero-shot generalization across diverse linguistic inputs. Additionally, we show that traditional evaluation metrics, such as Exact Match, often fail to assess semantic correctness. To remedy this, we introduce large language models (LLMs)-Judged Semantic Equivalence (SE), a new metric that leverages LLMs to evaluate whether generated formulas preserve the intended meaning. Extensive experiments demonstrate that MA-FOL outperforms strong baselines on Multi-MALLS, without any multilingual fine-tuning. The SE metric further reveals semantic correctness that traditional metrics miss. Overall, our work provides a benchmark to test robustness of NL2FOL, a framework to improve it, and a metric to evaluate it more effectively.

1 Introduction

Translating natural language into First-Order Logic (NL2FOL) is a long-standing and foundational task, serving as a critical component for various logic-based natural language processing (NLP) applications such as textual entailment and natural language reasoning. The field has progressed from early rule-based systems (Zettlemoyer and Collins, 2005; Abzianidze, 2017; Barker-Plummer et al., 2009) to neural sequence-to-sequence models and, more recently, paradigms leveraging the power of large language models (LLMs) (Ouyang et al., 2022; Brown et al., 2020; Radford et al., 2018; Devlin et al., 2019; Touvron et al., 2023). While recent advances in LLMs have improved NL2FOL translation performance, most existing models and benchmarks are predominantly focused on English. These systems are typically trained or fine-tuned on English-only corpora and implicitly learn English-specific syntactic and lexical patterns. As a result, they struggle to generalize across languages, exposing a critical weakness in terms of **cross-lingual robustness**. Moreover, current evaluation datasets rarely test a model's behavior under linguistic variation, and there is a lack of dedicated benchmarks to assess multilingual generalization.

To better understand and address the above-mentioned challenges, we construct **Multi-MALLS**, the first multilingual benchmark for NL2FOL in the community. Multi-MALLS extends the high-quality English MALLS dataset (Yang et al., 2023) to five additional languages, providing a diverse evaluation bed for measuring multilingual generalization. Then, we conduct the first systematic study on multilingual robustness for NL2FOL. Our investigation reveals that models trained or fine-tuned on English data experience significant performance degradation when tested on textual inputs from other languages. These findings suggest that the existing paradigm for NL2FOL modeling lacks robustness in real-world, multilingual scenarios.

To fill this gap,We propose a fully **training-free**, modular framework named the Multi-Agent

Framework for First-Order Logic (MA-FOL). Unlike previous approaches that rely on monolithic end-to-end training, MA-FOL operates entirely via prompt-based reasoning and requires no fine-tuning or supervision in the target language. It decomposes the NL2FOL task into three specialized modules: A Skeleton Generator, which extracts the language-agnostic logical structure of the input sentence. A Predicate Combiner, which grounds the structure with language-specific predicates and constants. A Dynamic Optimizer, which refines and canonicalizes the resulting FOL expression for logical consistency and readability. This modular design enables MA-FOL to perform *zero-shot cross-lingual generalization* by decoupling structural reasoning, lexical variation, and post-hoc refinement.

We also observe that standard metrics such as Exact Match (EM) are insufficient for assessing semantic correctness. These metrics penalize any deviation from a reference output, even if the generated FOL is logically equivalent. To solve this, we propose a novel metric called **LLM-Judged Semantic Equivalence (SE)**, which uses powerful LLMs to compare the meaning of the predicted and reference FOL expressions in the context of the original NL input.

In summary, our contributions are threefold:

- We construct **Multi-MALLS**, the first multilingual benchmark for evaluating NL2FOL robustness across six languages.
- We propose **MA-FOL**, a novel training-free multi-agent framework that improves zero-shot cross-lingual performance via modular reasoning.
- We introduce **LLM-Judged Semantic Equivalence (SE)**, a more faithful evaluation metric that captures the semantic correctness of logical outputs beyond surface-form matching.

Extensive experiments demonstrate that **MA-FOL, despite requiring no training**, significantly **outperforms existing state-of-the-art models that are fine-tuned on English**, when evaluated on cross-lingual NL2FOL tasks in Multi-MALLS. The SE metric further reveals semantic accuracy that is often overlooked by traditional evaluation methods.

2 Related Work

Natural Language to First-Order Logic. NL2FOL translation has evolved from early grammar-based systems (Zettlemoyer and Collins, 2005; Bos and Markert, 2005) to neural sequence-to-sequence models (Dong and Lapata, 2016; Singh et al., 2020). More recently, LLMs have achieved strong results via fine-tuning (Xu et al., 2024; Yang et al., 2023) or by reframing the task as code generation (Liu, 2025). However, this progress has been overwhelmingly monolingual, with datasets like FOLIO (Han et al., 2022) and MALLS (Yang et al., 2023) being English-only. This leaves the critical challenge of cross-lingual generalization unaddressed, a gap we tackle with Multi-MALLS and MA-FOL.

Cross-Lingual Semantic Parsing. While cross-lingual transfer is a major theme in NLP, supported by foundational work in pretraining (Conneau and Lample, 2019) and standardized benchmarks (Siddhant et al., 2020; Ruder et al., 2021), efforts in semantic parsing have largely targeted shallower representations like SQL or SPARQL. Zero-shot generalization for highly expressive formalisms like FOL remains underexplored. To our knowledge, our work is the first to systematically evaluate and improve cross-lingual NL2FOL robustness without requiring multilingual supervision.

Evaluation Metrics for Logic-Based Parsing. Standard metrics such as Exact Match (EM) are often too brittle. Logical Equivalence (LE) (Yang et al., 2023) offers an improvement by comparing truth tables but still penalizes semantically valid reformulations (e.g., predicate merging). Recent work has demonstrated the promise of using LLMs as semantic evaluators (Gu et al., 2024). Inspired by this, we propose LLM-Judged Semantic Equivalence (SE) to better capture the semantic correctness of generated logic in context.

Prompt-Based Modular Reasoning with LLMs. Advanced prompting strategies, including Chain-of-Thought (Wei et al., 2022; Yao et al., 2023) and modular or multi-agent frameworks (Chen et al., 2024; Lalwani et al., 2024), have enhanced LLM reasoning by decomposing complex tasks. However, these techniques remain underexplored for structured generation tasks like NL2FOL, especially in multilingual settings. Our MA-FOL framework applies this decomposition principle, using prompt-driven agents to separate language-agnostic logical structure from language-specific predicates, thereby enhancing zero-shot cross-lingual robustness.

Dataset	Source	# Pairs	Languages	FOL Operator Counts							
				$\forall$	$\exists$	$\neg$	$\wedge$	$\vee$	$\rightarrow$	$\leftrightarrow$	$\oplus$
FOLIO	Expert-written	∼2k	English	1,111	182	421	631	167	1,137	17	121
LogicNLI	Synthetic	∼12k	English	2,783	5,327	10,230	6,590	2,373	8,712	3,288	0
MALLS	Machine-gen.	34k	English	32,865	2,036	4,567	30,143	6,402	30,667	3,726	2,150
Multi-MALLS (Ours)	**Machine-gen. + Human Check**	**34k × 6**	**Multiple (6)**	**32,865**	**2,036**	**4,567**	**30,143**	**6,402**	**30,667**	**3,726**	**2,150**

Table 1: Comparison of Multi-MALLS with other NL2FOL benchmarks, including counts of FOL operators.

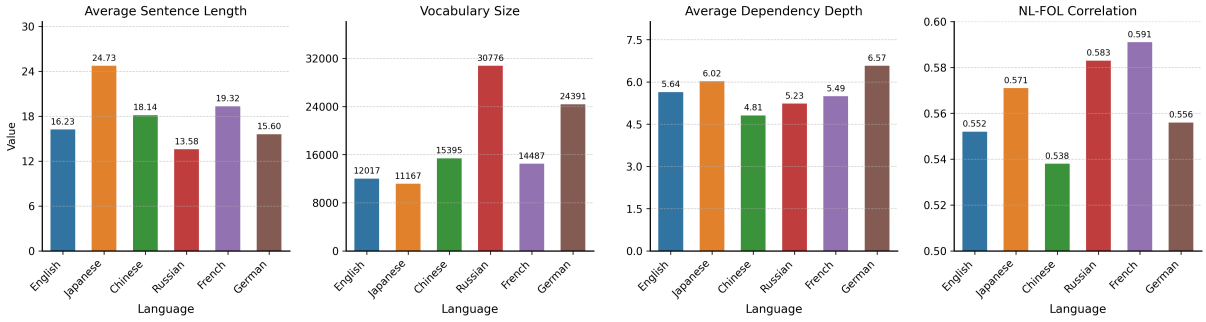

Figure 1: Statistical analysis of the Multi-MALLS dataset across languages.

3 The Multi-MALLS Dataset

3.1 Overview

To facilitate research on cross-lingual NL2FOL translation, we introduce Multi-MALLS. The core design principle of Multi-MALLS is to create a "many-to-one" mapping, where natural language sentences from multiple languages correspond to a single, canonical FOL expression. This setup provides an ideal testbed for evaluating a model's ability to remain robust to different linguistic surfaces while preserving a common logical ground.

3.2 Dataset Construction with Human Refinement

Our construction process is grounded in the high-quality English NL-FOL pairs from the MALLS dataset (Yang et al., 2023). To ensure the multilingual data maintains high linguistic quality and avoids the "translationese" often found in raw machine translation, we implemented a rigorous **Human-in-the-Loop** pipeline. The process is as follows:

1. **Source Data**: We utilize the original English NL-FOL pairs from MALLS as the semantic anchor.

2. **Step 1: MT Initialization**: We utilized `qwen3-30b` via the DashScope API as an initialization step to obtain baseline translations. We employed a context-aware prompting strategy that included the logical form to guide the model towards semantically accurate translations.

3. **Step 2: Expert Verification and Refinement**: To mitigate machine translation artifacts, we employed native-speaker experts (qualifications detailed in Appendix A.1) to review and polish the MT outputs. Experts were instructed to perform two specific tasks:

   - **Polishing for Fluency**: Reviewers corrected grammatical errors and refined unnatural phrasing to ensure the sentences sound fluent to native speakers.
   - **Logic Verification**: Reviewers verified that the translated sentence strictly aligns with the truth conditions of the original First-Order Logic expression.

   This human refinement step ensures that the dataset captures correct semantics without being limited by the rigidity of raw machine translation.

4. **Step 3: Quality Control**: Any samples containing ambiguous translations or factual errors that could not be easily corrected were discarded to maintain the benchmark's reliability.

3.3 Dataset Statistics

Table 1 shows that while previous benchmarks (Tian et al.; Yang et al., 2023; Han et al., 2022) are exclusively in English, Multi-MALLS introduces five additional languages, creating a large-scale parallel corpus crucial for evaluating models with robust cross-lingual capabilities.

3.4 Analysis Methods

To quantify the linguistic characteristics of Multi-MALLS, we employed a series of standard automated analysis methods (Jurafsky and Martin, 2023), primarily based on the spaCy library (Honnibal et al., 2020) and its pre-trained models for each respective language.

- **Sentence Length** for most languages was defined as the number of words after removing punctuation. Considering the characteristics of their writing systems, we used the total number of tokens for Chinese and Japanese.

- **Vocabulary Size** was determined by counting the number of unique, case-insensitive, non-punctuation tokens in the entire corpus for each language.

- **Syntactic Complexity** was measured by the average dependency depth. We used spaCy's dependency parser, which is based on the Universal Dependencies framework (Nivre et al., 2016), to generate a syntactic tree for each sentence and then calculated the maximum path length from the root of the tree to its furthest leaf node. This metric reflects the grammatical nesting and structural complexity of the sentences.

- **NL-FOL Correlation** was calculated to verify the semantic consistency of the dataset. We computed the Pearson correlation coefficient (James et al., 2013) between sentence length and FOL complexity. Here, FOL complexity was quantified as the number of predicates in each logical expression.

3.5 Dataset Analysis

The results of our analysis are summarized in Figure 1. They reveal several key insights into the dataset's nature and the challenges it poses.

First, the dataset exhibits significant diversity in surface-form properties. For instance, the average sentence length in Japanese (24.73) is nearly double that of Russian (13.58), and the vocabulary size in Russian (30,776) is substantially larger than in other languages, likely due to its rich morphology. This demonstrates that a robust model must handle considerable variations in verbosity and lexical diversity.

Second, the syntactic complexity, measured by average dependency depth, also varies widely. German (6.57) and Japanese (6.02) show the most complex grammatical structures on average, while Chinese (4.81) is the most syntactically concise. This diversity in syntactic structure poses a significant challenge for parsing and motivates the design of our *Skeleton Generator*, which must extract a common logical structure from these varied inputs.

Finally, and most importantly, the correlation between sentence length and FOL complexity is consistently positive and stable across all languages (ranging from 0.538 to 0.591). This indicates that our translation pipeline successfully preserved the core relationship between linguistic expression and logical complexity. This consistency validates Multi-MALLS as a high-quality and reliable benchmark for evaluating the semantic understanding and robustness of NL2FOL models.

4 MA-FOL: A Modular Framework for Cross-Lingual Robustness

4.1 Core Idea: Decoupling and Refinement

To address the challenge of cross-lingual generalization in NL2FOL translation, we propose **MA-FOL** (Multi-Agent Framework for FOL), a novel, training-free modular framework. Existing models often exhibit a sharp performance degradation on non-English inputs, a problem rooted in the entanglement of language-specific syntactic and lexical patterns with language-agnostic logical structures.

The core design principle of MA-FOL is **decoupling**. It employs a "generate-then-refine" strategy (illustrated in Figure 2) to decompose the complex translation process into complementary sub-tasks:

1. **Language-Agnostic Logical Structuring**: Identifying the abstract logical relationships independent of any specific language.

2. **Language-Specific Lexical Grounding**: Mapping concrete semantic meanings into the abstract logical structure.

The framework's design is inspired by recent work demonstrating the efficacy of in-context learning

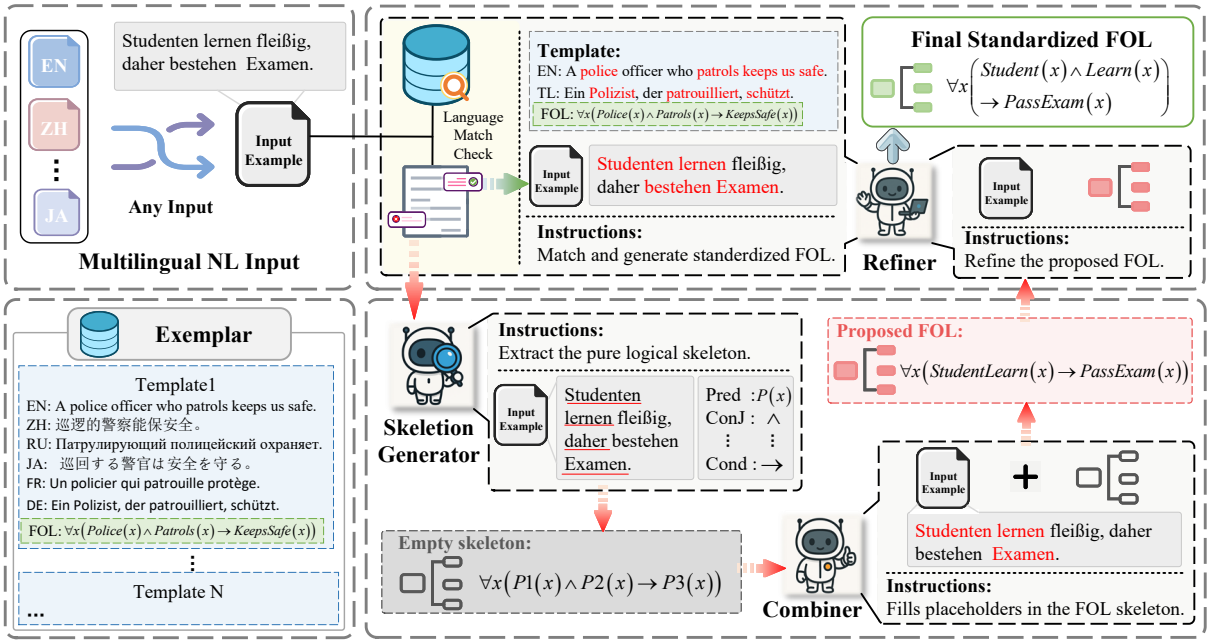

Figure 2: The architecture of our Multi-Agent Framework (MA-FOL). The process is decomposed into specialized components designed to handle different aspects of linguistic variation, enhancing cross-lingual robustness. MA-FOL adopts a fully training-free pipeline with three modules: the Skeleton Generator for extracting logical form, the Combiner for grounding predicates, and the Exemplar-based Refiner for template-based optimization and standardization.

and code generation for solving complex symbolic transformation tasks without fine-tuning.

4.2 Stage 1: Multi-Agent Generation

In the first stage, we deploy two specialized agents to handle the two primary sources of cross-lingual variance: syntactic and lexical.

Component 1: Logical Skeleton Generator This component is designed to overcome **syntactic variation** across languages. Whether the input is English, "Every dog barks", or German, "Jeder Hund bellt", the underlying logical structure remains identical. This agent's task is to extract this universal, language-agnostic logical "skeleton" and represent it as a templated logical form with placeholders.

- **Example**: For the inputs above, it should consistently generate $\forall x(P1(x) \rightarrow P2(x))$. This skeleton serves as a robust intermediate representation, providing a unified structural foundation for the subsequent grounding step.

Component 2: Predicate Combiner This component focuses on resolving **lexical variation**. Its task is to bind the placeholders from the previous stage (e.g., $P1, P2$) with concrete semantic concepts extracted from the original sentence. To construct a unified semantic space, this agent performs **canonicalization** on all predicates, mapping them to a standardized English `UpperCamelCase` format.

- **Example**: A complex concept expressed in Chinese as "shuō duōzhǒng yǔyán de rén" (a person who speaks multiple languages) is canonicalized to the single predicate `SpeaksMultipleLanguages(x)`. This ensures that diverse linguistic expressions are mapped to the same logical predicate.

4.3 Stage 2: Exemplar-based Refiner

After the initial First-Order Logic (FOL) expression, denoted as F_{gen}, is generated, we introduce an advanced refinement stage that acts as a **consistency and accuracy debugger**. This stage uses an exemplar-based approach to verify and optimize the preliminary output.

Template Library Construction and Matching The refiner operates by matching the generated formula against a lightweight, data-driven template library. To construct this library, we randomly sample 10 instances from the English MALLS training set. For each instance, we manually abstract its ground-truth FOL into a canonical logical structure, preserving its core operators and patterns.

This process results in a small but powerful set of high-quality exemplars, which we call the template library $\mathcal{P}$.

Given a generated formula F_{gen}, the refiner first extracts its abstract logical structure, denoted as Structure(F_{gen}). It then checks if this structure matches any of the trusted patterns in the library $\mathcal{P}$.

Decision and Refinement Logic The core logic of the refiner is to decide whether to trust the initial generation or to correct it based on a known, high-quality pattern. The final output, F_{final}, is determined by the following rule:

$$F_{\text{final}} = \begin{cases} \text{Refine}(F_{\text{gen}}), & \text{Structure}(F_{\text{gen}}) \in \mathcal{P}_{\text{select}} \\ F_{\text{gen}}, & \text{otherwise} \end{cases} \quad (1)$$

This logic is implemented through two main pathways:

- **Consistency Enforcement**: If the structure of the generated formula, Structure(F_{gen}), **matches** a trusted pattern in the selected set $\mathcal{P}_{selected}$, the framework invokes a refinement function, Refine(F_{gen}). This function regenerates the formula according to the high-quality template, ensuring the output adheres to a canonical and logically complete form.

- **Exploratory Adoption**: If the structure of F_{gen} does **not match** any known pattern, it may represent a novel logical construction. In this case, the framework pragmatically **adopts** the initial generation. This mechanism allows MA-FOL to handle new linguistic phenomena flexibly while enforcing consistency for common patterns.

5 Experiments

5.1 Experimental Setup

Dataset. To rigorously evaluate cross-lingual robustness, all models were fine-tuned exclusively on the English MALLS training set. Evaluation was performed on a 1,000-sample test set from our Multi-MALLS benchmark.

Baselines. We compare MA-FOL against several strong baselines: **LOGICLLAMA** (Yang et al., 2023), a LLaMA-7B model fine-tuned with SFT+RLHF for FOL translation; **Symbol-LLM** (Xu et al., 2024), a LLaMA-2 model tuned on a wide range of symbolic tasks; **CODE4LOGIC** (Liu, 2025), a training-free method that reframes the task as code generation; and a powerful generic LLM (i.e., **Qwen-Max** (Yang et al., 2025)) using few-shot prompting.

Evaluation Metrics. We use two metrics. The first, **Logical Equivalence (LE)** (Yang et al., 2023), compares truth tables to check for logical equivalence and is more robust than Exact Match. However, LE can be overly strict, penalizing semantically correct but syntactically divergent outputs . To illustrate this limitation, consider the following example for the sentence "A red apple is sweet and a green apple is sour, while a yellow apple is a balance between sweet and sour.":

Ground Truth:

$$\forall x \forall y \forall z \big((\texttt{RedApple}(x) \land \texttt{GreenApple}(y) \land \texttt{YellowApple}(z)) \rightarrow ((\texttt{IsSweet}(x) \land \texttt{IsSour}(y)) \land (\texttt{IsSweet}(z) \oplus \texttt{IsSour}(z))) \big)$$

MA-FOL Prediction:

$$\forall x(\texttt{RedApple}(x) \rightarrow \texttt{Sweet}(x)) \land \forall x(\texttt{GreenApple}(x) \rightarrow \texttt{Sour}(x)) \land \forall x(\texttt{YellowApple}(x) \rightarrow \texttt{BalancedSweetAndSour}(x))$$

Here, our model's output is arguably a more faithful semantic interpretation, correctly identifying the sentence as three independent, universal statements about classes of apples. Despite its correctness, its LE score is 0.0 because its logical structure diverges from the reference. This case perfectly demonstrates the need for a metric that can look beyond syntactic form.

To address this, our second, proposed metric is **LLM-Judged Semantic Equivalence (SE)**. We define SE as a formal, hybrid evaluation protocol that rewards semantic correctness while retaining a lexical base. The calculation proceeds as follows:

1. **Baseline Score Calculation:** For each sample, we first compute its **Logical Equivalence (LE) score**. This score serves as the default or baseline score for the sample.

2. **Semantic Judgment by Arbitrator:** We employ a powerful, third-party LLM (Qwen3-235b-a22b)(Yang et al., 2025) as an impartial semantic judge. The judge is presented with the source NL, the ground-truth FOL, and the

Language	LOGICLLAMA-7B		Symbol-LLM-7B		Symbol-LLM-13B		Qwen-Max (FS)		MA-FOL (Ours)	
	LE	SE	LE	SE	LE	SE	LE	SE	LE	SE
English (EN)	0.8937	0.9233	**0.9104**	**0.9358**	0.9049	0.9350	0.7592	0.8339	0.8373	0.9093
French (FR)	**0.8586**	0.8667	0.8211	0.8354	0.7939	0.8134	0.7420	0.7724	0.8255	**0.8782**
German (DE)	**0.8447**	0.8505	0.8210	0.8337	0.7625	0.7796	0.7108	0.7384	0.8221	**0.8705**
Russian (RU)	0.8175	0.8250	0.7239	0.7480	0.5088	0.5341	0.7628	0.8096	**0.8353**	**0.8702**
Chinese (ZH)	0.8349	0.8372	0.5219	0.5567	0.4724	0.5037	0.7710	0.8023	**0.8395**	**0.8906**
Japanese (JP)	0.8021	0.8051	0.5229	0.5576	0.4691	0.4988	0.7668	0.8105	**0.8326**	**0.8799**
Final Score	0.8466		0.7324		0.6647		0.7733		**0.8584**	

Table 2: Main results for the zero-shot cross-lingual evaluation on the Multi-MALLS test set. The baselines consist of models **fine-tuned on English data** (LOGICLLAMA, Symbol-LLM) and a large language model performing **English few-shot (FS) prompting** (Qwen-Max). Our MA-FOL framework uniquely leverages **multilingual examples** in its prompts to enhance robustness. Its superior performance is evident in the LE (Logical Equivalence) scores and further amplified by the SE (LLM-Judged Semantic Equivalence) scores. A 'Final Score' (the average of Average LE and SE) provides a single metric for model comparison.

predicted FOL, and must classify the prediction into one of three categories: 'equivalent', 'prediction_better', or 'truth_better'.

3. **Conditional Score Replacement:** A conditional replacement mechanism is then triggered. If the judgment is 'equivalent' or 'prediction_better', the sample's baseline LE score is **replaced** by a dedicated **'agentscore'**, which is set to 1.0 to maximally reward the semantically correct generation. If the judgment is 'truth_better', the original LE score is retained.

4. **Final Score Aggregation:** The final SE score reported in our results is the **average** of the final scores for all samples in the test set (a mix of original LE scores and the rewarding 'agentscore' of 1.0).

To validate the reliability of our LLM-based judge, we conducted a human-LLM agreement study. We randomly sampled 100 instances from our test set, covering all languages. Two human experts with backgrounds in logic and linguistics were asked to perform the same three-way classification task as the LLM judge. The LLM's judgments achieved a 95% agreement rate with the consensus of the human experts, with a Cohen's Kappa of 0.91. This high level of agreement confirms that our LLM judge serves as a reliable proxy for human evaluation, validating the credibility of the SE metric. The SE protocol combines the formal LE score as a rigorous baseline with agent judgment as a flexible reward signal, encouraging outputs that capture true semantic meaning even when they diverge syntactically from the ground truth.

5.2 Main Results and Analysis

Our main experimental results, presented in Table 2, offer compelling evidence for the superior cross-lingual robustness of our proposed MA-FOL framework. We analyze these findings from three key perspectives:

Baselines' Brittleness Confirms the Core Challenge. As hypothesized, all baseline models, which are fine-tuned exclusively on English data, exhibit a significant performance degradation when evaluated on non-English languages. The performance drop is particularly stark for models like Symbol-LLM on languages syntactically distant from English, such as Chinese and Japanese. For example, Symbol-LLM-7B's LE score plummets from 0.9104 on English to 0.5219 on Chinese. This confirms our initial hypothesis that monolithic fine-tuning causes models to overfit to the syntactic and lexical patterns of the source language, rendering them *brittle* against linguistic variation.

MA-FOL's Consistent Performance Demonstrates True Robustness. In stark contrast, MA-FOL maintains remarkably stable and high performance across all languages. Its 'Final Score' (0.8584) surpasses all baselines, including the strong LOGICLLAMA (0.8466). This consistency is a direct result of our modular design. The **Skeleton Generator** successfully abstracts away surface-level linguistic differences (e.g., word order, inflection), while the **Predicate Combiner** canonicalizes diverse lexical expressions into a unified semantic space. This decoupling strategy is the cornerstone of MA-FOL's robustness.

The SE Metric Reveals Deeper Semantic Understanding. Crucially, the Semantic Equivalence (SE) metric reveals insights that Logical Equivalence (LE) alone cannot. Across all languages, the gap between SE and LE scores is consistently larger for MA-FOL than for the baselines. This indicates that MA-FOL frequently generates logical forms that are semantically correct and often more concise (e.g., composing 'Effective' and 'Vaccine' into 'EffectiveVaccine'), even if they don't exactly match the ground truth syntax. This ability for intelligent semantic composition is a hallmark of a truly robust system, and the SE metric is vital for capturing it.

5.3 Ablation Study

To dissect the source of MA-FOL's robustness, we conducted an ablation study (Table 3). The results clearly demonstrate the synergistic contribution of each component.

The Necessity of the Refiner. Removing the Exemplar-based Refiner (`w/o Refiner`) causes a significant drop in performance. This variant often produces structurally sound but incomplete formulas, a flaw highlighted in our Case Study where the domain predicate `Product(x)` might be missed. This confirms the refiner's critical role as a final-stage verifier for logical completeness and consistency.

The Value of Modular Generation. Removing the multi-agent generation stage (`w/o Multi-Agent`) and relying on a direct prompt to the LLM is still inferior to the full framework. This shows that while a powerful LLM has strong zero-shot capabilities, the structured decomposition into skeleton generation and predicate grounding provides essential scaffolding that guides the model away from common structural errors and improves overall precision.

Configuration	LE	SE
w/o Refiner	0.7810	0.8125
w/o Multi-Agent	0.8254	0.8772
MA-FOL (Full)	**0.8321**	**0.8831**

Table 3: Averaged ablation study results across languages, comparing model configurations on LE and SE metrics.

5.4 Case Study

To illustrate our framework's strengths, we analyze a challenging Japanese sentence from MultiMALLS whose logic hinges on a three-part disjunction. The ground truth FOL is:

$$\begin{aligned}&\forall x(\texttt{Product}(x) \quad \wedge\\ &(\texttt{SustainableMaterials}(x) \quad \vee\\ &\texttt{LowCarbonFootprint}(x) \quad \vee\\ &\texttt{EncouragesConservation}(x)) \quad \rightarrow\\ &\texttt{EcoFriendly}(x))\end{aligned}$$

A comparison reveals the brittleness of baselines: **LOGICLLAMA-7B** mistook the disjunction for a conjunction ($\wedge$); **Symbol-LLM-7B** failed completely, outputting natural language; **Symbol-LLM-13B** recognized the disjunction but failed to structure the three predicates correctly; and **Qwen-Max (FS)** generated incomplete logic by omitting the crucial domain predicate `Product(x)`.

In contrast, **MA-FOL (Ours)** was the only method to produce a logically perfect translation, correctly identifying both the domain predicate and the three-part disjunction:

$$\begin{aligned}&\forall x(\texttt{Product}(x) \quad \wedge\\ &(\texttt{MadeOfSustainableMaterials}(x) \quad \vee\\ &\texttt{HasLowCarbonFootprint}(x) \quad \vee\\ &\texttt{PromotesResourceConservation}(x)) \rightarrow \ldots)\end{aligned}$$

6 Conclusion

In this paper, we address the critical challenge of cross-lingual robustness in NL2FOL translation. We introduce three core contributions: the **Multi-MALLS** benchmark for multilingual evaluation; the training-free, modular **MA-FOL** framework, which achieves superior zero-shot performance; and the **LLM-Judged Semantic Equivalence (SE)** metric for a more faithful evaluation. Our extensive experiments show that MA-FOL significantly outperforms strong, English-fine-tuned baselines, demonstrating a new path towards building truly robust language-to-logic systems. The success of our approach stems from its core principle: decoupling language-agnostic logical structuring from language-specific lexical grounding.

Our findings support a shift in semantic parsing to modular, interpretable frameworks, which offer greater robustness and control than brittle end-to-end models.

7 Limitations

Our work, while providing a new benchmark and a robust framework for cross-lingual NL2FOL, has

several limitations that offer avenues for future research.

First, the performance of our training-free **MA-FOL** framework is intrinsically dependent on the capabilities of the underlying large language model used for its agentic components. While it demonstrates strong zero-shot generalization, its success is tied to the reasoning power of the backbone LLM, and its performance may vary with different models. Furthermore, its multi-agent pipeline, which requires sequential LLM calls, incurs higher latency and computational costs compared to a single forward pass of a fine-tuned model. The exemplar-based refiner also relies on a small, fixed set of 10 templates, which may not cover all valid logical structures, potentially failing to correct errors in novel constructions.

Second, our **Multi-MALLS** benchmark, while being the first of its kind, is constructed via machine translation from an English source. Despite human verification for logical alignment and fluency, the resulting sentences may exhibit "translationese" and might not fully capture the idiomatic and syntactic diversity of natively authored sentences in the target languages. Additionally, the domain and complexity of the NL-FOL pairs are inherited from the original MALLS dataset and may not encompass more complex, real-world reasoning scenarios.

Finally, our proposed **SE** metric's reliability is contingent on the LLM judge. Although we validate its performance with a human-agreement study, the metric's ultimate accuracy is bounded by the judge's capabilities. The LLM may still have inherent biases (e.g., favoring certain logical forms) or fail to grasp subtle nuances in highly complex logical formulas, potentially leading to evaluation errors.

References

Lasha Abzianidze. 2017. LangPro: Natural language theorem prover. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 115–120, Copenhagen, Denmark. Association for Computational Linguistics.

Dave Barker-Plummer, Richard Cox, and Robert Dale. 2009. Dimensions of difficulty in translating natural language into first-order logic. In *Educational Data Mining - EDM 2009, Cordoba, Spain, July 1-3, 2009. Proceedings of the 2nd International Conference on Educational Data Mining*, pages 220–229. www.educationaldatamining.org.

Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 628–635.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Pei Chen, Shuai Zhang, and Boran Han. 2024. Comm: Collaborative multi-agent, multi-reasoning-path prompting for complex problem solving. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1720–1738.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. *Advances in neural information processing systems*, 32.

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.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43, Berlin, Germany. Association for Computational Linguistics.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Luke Benson, Lucy Sun, Ekaterina Zubova, Yujie Qiao, Matthew Burtell, and 1 others. 2022. Folio: Natural language reasoning with first-order logic. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4084–4103.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Allison Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. In *17th Python in Science Conference*.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An introduction to statistical learning*, volume 112. Springer.

Dan Jurafsky and James H Martin. 2023. *Speech and language processing*, 3rd edition. Prentice Hall.

Abhinav Lalwani, Lovish Chopra, Christopher Hahn, Caroline Trippel, Zhijing Jin, and Mrinmaya Sachan. 2024. Nl2fol: Translating natural language to first-order logic for logical fallacy detection. *arXiv preprint arXiv:2405.02318*.

Junnan Liu. 2025. Few-shot natural language to first-order logic translation via code generation. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10939–10960. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan Mcdonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, and 1 others. 2016. Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training. *OpenAI Blog*.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10215–10245, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Aditya Siddhant, Junjie Hu, Melvin Johnson, Orhan Firat, and Sebastian Ruder. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In *Proceedings of the International Conference on Machine Learning*, volume 2020, pages 4411–4421.

Hrituraj Singh, Milan Aggrawal, and Balaji Krishnamurthy. 2020. Exploring neural models for parsing natural language into first-order logic. *arXiv preprint arXiv:2002.06544*.

Jidong Tian, Yitian Li, Wenqing Chen, Liqiang Xiao, Hao He, and Yaohui Jin. Diagnosing the first-order logical reasoning ability through logicnli. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3738–3747.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Fangzhi Xu, Zhiyong Wu, Qiushi Sun, Siyu Ren, Fei Yuan, Shuai Yuan, Qika Lin, Yu Qiao, and Jun Liu. 2024. Symbol-LLM: Towards foundational symbol-centric interface for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13091–13116, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Yuan Yang, Siheng Xiong, Ali Payani, Ehsan Shareghi, and Faramarz Fekri. 2023. Harnessing the power of large language models for natural language to first-order logic translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6942–6959, Toronto, Canada. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

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

A Dataset Details

Data Samples To provide concrete illustrations of the translation challenges and our model's capabilities, we present two representative examples selected from the Multi-MALLS test set. Figure 3 shows a simple conditional structure, while Figure 4 demonstrates a more complex biconditional case involving disjunction.

A.1 Human Verification Protocol

While our FOL-aware prompting strategy yielded high-quality translations, we incorporated a crucial human verification step to ensure the final dataset's reliability. This process was conducted by language experts for each of the five target languages.

Verifier Qualifications. Verifiers were native speakers of the target language with a university-level education. While formal training in logic was not required, they were selected based on their high proficiency in English and their ability to understand nuanced semantic instructions.

Verification Guidelines. Each verifier was presented with the original English sentence, the ground-truth FOL expression, and the machine-translated sentence. They were given the following two primary instructions:

1. **Assess Fluency and Naturalness:** Is the translated sentence grammatically correct, fluent, and natural-sounding in the target language? Or does it exhibit "translationese"?

2. **Verify Semantic and Logical Alignment:** Does the translated sentence's meaning perfectly align with the logical constraints imposed by the FOL formula? For example, if the FOL specifies a universal quantifier ($\forall$), does the translation correctly express "all" or "every"? If the FOL uses a disjunction ($\vee$), does the translation accurately reflect an "or" relationship?

Correction and Discarding Protocol. Verifiers were instructed to directly correct minor issues such as typos or awkward phrasing. If a translation was found to be semantically ambiguous, factually incorrect, or logically misaligned with the FOL, it was flagged. Flagged sentences were then reviewed by a senior linguist. If the issue could not be resolved with a simple correction, the entire data sample (including all its translations) was discarded from the final dataset to maintain high quality. Approximately 1.2% of the initial machine-translated samples were discarded during this process.

Example of Translation Quality Verification To ensure the quality and logical fidelity of the Multi-MALLS dataset, each translation underwent a verification process. This section provides a concrete example of this process. The ground truth is established by the original English sentence and its corresponding FOL representation.

> **English NL:** "A gemstone can be a diamond, a ruby, or an emerald, but not more than one type of gemstone." **Ground Truth FOL:**
>
> $\forall x$ (Gemstone(x) $\rightarrow$ ((Diamond(x) $\wedge$ $\neg$(Ruby(x) $\vee$ Emerald(x))) $\vee$ ...))

Our analysis revealed issues ranging from minor grammatical imperfections to significant factual errors. For this specific example, the most critical issue was a factual mistranslation in the Chinese version, where "emerald" (Pinyin: zǔmǔlǔ) was incorrectly translated as "sapphire" (Pinyin: lánbǎoshí). A minor grammatical error was also noted in the French translation. The findings are summarized in Table 4.

B Detailed Case Study: Japanese Disjunctive Logic

To provide a concrete illustration of our framework's strengths, we analyze a challenging Japanese sentence from the Multi-MALLS test set. This example is particularly insightful as its core logical structure hinges on a three-part **disjunction (OR logic)**, a common feature in natural language that proves difficult for many models.

The input sentence is a conditional definition:

> **Japanese NL:** *Seihin ga jizoku kanō na sozai de tsukurare, tei-tanso futtopurinto o mochi, matawa shigen no hogo o sokushin suru baai, sono seihin wa eko-furendorī desu.*
>
> **English Translation:** A product is eco-friendly if it is made from sustainable materials, has a low carbon footprint, or encourages conservation of resources.
>
> **Ground Truth FOL:** $\forall x$(Product(x) $\wedge$ (SustainableMaterials(x) $\vee$ LowCarbonFootprint(x) $\vee$ EncouragesConservation(x)) $\rightarrow$ EcoFriendly(x))

We compare the outputs of several baseline models against our MA-FOL framework in Figure 5.

The comparison shown in Figure 5 is striking. The fine-tuned models either fundamentally misunderstood the core logical operator (‘and‘ vs.

Language	Natural Language / First-Order Logic
Example 1: Simple Conditional	
English	A vacation is relaxing if it includes beautiful scenery and enjoyable activities.
Japanese	休暇は、美しい風景と楽しむことができる活動を含んでいる場合、リラックスしています。
Chinese	如果一次假期包含美丽的风景和愉快的活动，那么它就是令人放松的。
Russian	Вакация является расслабляющей, если она включает красивые пейзажи и приятные занятия.
French	Une vacance est relaxante si elle inclut un paysage magnifique et des activités agréables.
German	Ein Urlaub ist entspannend, wenn er schöne Landschaften und unterhaltsame Aktivitäten beinhaltet.
FOL	$\forall$x (Vacation(x) $\wedge$ Relaxing(x) $\rightarrow$ (BeautifulScenery(x) $\wedge$ EnjoyableActivities(x)))

Figure 3: Test Set Example 1: A simple conditional structure. This example tests the model's ability to correctly parse the 'if...then' relationship across different languages.

Table 4: Summary of the quality verification for the "gemstone" example. This process was crucial for correcting errors and ensuring the high quality of the final dataset.

Language	Quality Rating	Main Issue / Notes
Chinese	Low	**Factual Error**: "emerald" (Pinyin: zǔmǔlǔ) was mistranslated as "sapphire" (Pinyin: lánbǎoshí).
French	Medium-High	Minor grammatical issue (incorrect gender of the article for "rubis").
Japanese	High	Accurate and natural translation capturing the exclusive-or logic.
Russian	High	Accurate and natural translation.
German	High	Accurate and natural translation.

'or'), failed to parse the structure correctly, or failed the task entirely. The general-purpose LLM, Qwen-Max (FS), performed significantly better by correctly identifying the disjunctive structure. However, it still produced an incomplete formula by omitting the crucial domain-scoping predicate `Product(x)`, which specifies that this rule applies only to products.

In contrast, our **MA-FOL** framework produced a logically perfect translation. This case vividly demonstrates that MA-FOL's structured, multi-stage process—generation followed by verification and refinement—is the key to achieving a level of precision and logical completeness that surpasses both monolithic fine-tuned models and direct few-shot prompting of powerful LLMs.

C Implementation Details

C.1 Experimental Settings and Infrastructure

MA-FOL Framework Details The agentic components of our MA-FOL framework were implemented using the **Qwen-Max** large language model, accessed via the DashScope public API. For all generation tasks, we used a consistent parameter setting to ensure reproducibility: a `temperature` of **0.1** and a `top_p` of **0.9**. The average latency for processing a single sentence through the entire generate-then-refine pipeline was approximately 3.5 seconds on our testing infrastructure.

Baseline Models Setup The baseline models used for comparison were evaluated in a zero-shot or few-shot setting without any additional fine-tuning on our dataset. For model-based baselines

Example 2: Biconditional with Disjunction	
English	To be eligible for a promotion, an employee must work for two years and have good performance, unless they have a supervisor's recommendation.
Japanese	昇進資格があるためには、従業員は少なくとも2年間会社で勤務し、良いパフォーマンス記録を持っている必要があり、上司からの推薦がある場合は除く。
Chinese	要符合晋升资格，员工必须在公司工作至少两年并且有良好的绩效记录，除非他们有一名主管的推荐。
Russian	Чтобы быть кандидатом на повышение, сотрудник должен проработать не менее двух лет и иметь хороший рейтинг, за исключением случаев с рекомендацией.
French	Pour être éligible à une promotion, un employé doit avoir travaillé deux ans et avoir un bon historique, à moins qu'il n'ait une recommandation.
German	Um qualifiziert zu sein, muss ein Mitarbeiter zwei Jahre gearbeitet haben und eine gute Leistung haben, es sei denn, er hat eine Empfehlung.
FOL	$\forall$x (Employee(x) $\wedge$ EligibleForPromotion(x) $\leftrightarrow$ ((WorkedAtCompanyForTwoYears(x) $\wedge$ GoodPerformanceRecord(x)) $\vee$ RecommendationFromSupervisor(x)))

Figure 4: Test Set Example 2: A more complex biconditional structure with a disjunctive (‘or‘) clause. This case challenges the model’s understanding of multiple logical operators and sentence structure.

such as **LOGICLLAMA** and **Symbol-LLM**, we utilized the publicly available, pre-trained model checkpoints provided by their original authors to perform direct inference on our Multi-MALLS test set. For prompting-based methods such as our **Qwen-Max (FS)** baseline, we followed the few-shot prompting methodologies described in their respective papers, using examples from the English training set as in-context demonstrations. This approach ensures a fair comparison of zero-shot cross-lingual generalization capabilities against our entirely training-free MA-FOL framework. All experiments were conducted on a server equipped with 4 NVIDIA A100 (40GB) GPUs.

C.2 Agent Prompts

This section details the prompts used in our framework.

Dataset Construction Prompt Our Multi-MALLS dataset was generated using a programmatic pipeline. A key component was the FOL-aware prompting strategy. Figure 6 shows the detailed instructions given to the LLM to generate high-quality, logic-preserving translations.

MA-FOL Framework and SE Metric Prompts This section presents the detailed prompts used for each agent in our framework and for the SE metric evaluation (Figures 7-10).

Table 5: Hyperparameter settings. Constraints were kept consistent across all modules to ensure fair comparison.

Parameter	Value
Model Configurations	
Primary Backbone Model	Qwen-Max (via DashScope API)
Open-Source Model	CodeLlama-13b-Instruct-hf
Temperature	0.1
Max Output Tokens	512
MA-FOL Framework Settings	
Refiner Template Pool Size	10 (randomly sampled)
Prompt Format	ChatML / Instruction Format

D Generalization to Open-Weights Models

To verify that the effectiveness of MA-FOL is not limited to the specific architecture or capability of flagship proprietary models (like Qwen-Max), we conducted additional experiments using the open-weights model **CodeLlama-13b-Instruct-hf**.

Model	Generated FOL / Output	Analysis of Key Logical Error
LOGICLLAMA-7B	`∀x (... ∧ ContainsSustainableMaterial(x) ∧ HasLowCarbonFootprint(x) ∧ PromotesResourceProtection(x) → ...)`	**Incorrect Operator**: Fundamentally misunderstood the sentence, mistaking the disjunctive 'or' relationship for a conjunctive 'and'.
Symbol-LLM-7B	`If the product is made from sustainable materials...` (Output was natural language)	**Catastrophic Failure**: Completely failed the task by not generating a formal logical expression. This indicates a lack of robustness.
Symbol-LLM-13B	`∀x (... ∧ SustainableMaterial(x) ∧ (LowCarbonFootprint(x) ∨ ResourceConservation(x)) → ...)`	**Incorrect Structure**: Partially recognized the 'or' relationship but failed to structure the three predicates correctly, resulting in a flawed logical form.
Qwen-Max (FS)	`∀x ((SustainableMaterial(x) ∨ LowCarbonFootprint(x) ∨ PromotesResourceConservation(x)) → EcoFriendly(x))`	**Incomplete Logic**: Correctly identified the 'or' relationship but omitted the key domain predicate `Product(x)`, making the rule overly general.
MA-FOL (Ours)	`∀x (Product(x) ∧ (MadeOfSustainableMaterials(x) ∨ HasLowCarbonFootprint(x) ∨ PromotesResourceConservation(x)) → ...)`	**Correct**. Successfully identified the complete logical structure, including the domain predicate and the three-part disjunction.

Figure 5: Case study comparison on a Japanese sentence with three-part disjunctive logic. This table highlights MA-FOL's superior ability to parse complex logical structures compared to strong baseline models.

```
You are a precise translation engine. Translate the 'Natural Language (NL)' text
    into the specified target language, ensuring the logical structure from the '
    First-Order Logic (FOL)' is preserved. Provide *only* the translated sentence
    itself, without any introductory phrases, labels, or the original text repeated.


Natural Language (NL): {nl_text}
First-Order Logic (FOL): {fol_representation}
Target Language: {target_language_name} ({target_language_code})

Please provide the translation of the NL sentence into {target_language_name} below:
```

Figure 6: Prompt for automated dataset translation.

We compared the performance of a standard **Single-Agent** baseline (direct prompting) against our **Multi-Agent (MA-FOL)** framework. As shown in Table 6, MA-FOL demonstrates strong generalization capabilities, improving the average Logical Equivalence (LE) score across most languages.

Significant Gains in Morphologically Rich Languages. The most striking improvement was observed in **Russian**, where the LE score surged from 0.5619 to 0.6847, representing a relative improvement of **21.9%**. Significant gains were also recorded in English (+9.0%) and German (+8.4%).

These results confirm that the "Decompose-Generate-Refine" paradigm of MA-FOL serves as a model-agnostic scaffolding strategy. It effectively unlocks the potential of smaller, open-source models by breaking down complex cross-lingual logic reasoning into manageable sub-tasks, rather than relying solely on the parametric knowledge of trillion-parameter models.

Language	Single Agent	MA-FOL (Ours)	Improvement
English (EN)	0.6482	**0.7068**	+9.0%
Russian (RU)	0.5619	**0.6847**	**+21.9%**
German (DE)	0.6214	**0.6736**	+8.4%
Chinese (ZH)	0.6340	**0.6751**	+6.5%
French (FR)	0.6174	**0.6374**	+3.2%
Japanese (JP)	**0.5741**	0.5344	-6.9%
Average	0.6095	**0.6520**	**+7.0%**

Table 6: Performance comparison (Logical Equivalence) on **CodeLlama-13b-Instruct-hf**. MA-FOL consistently boosts performance across most languages, with the most significant gains observed in Russian. Note: The regression in Japanese is likely due to the limited tokenizer support for mixed-script Japanese in the CodeLlama base model.

E Metric Validation with Third-Party Auditor

A potential threat to validity in LLM-based evaluation is "family bias," where a judge model (e.g., Qwen) might preferentially rate outputs generated by the same model family. To rigorously validate the objectivity of our **SE (Semantic Equiva-**

```
You are an expert multilingual logical analyst. Your mission is to parse a sentence,
     which could be in various languages, and extract its pure logical skeleton.
    Your goal is to see through the surface-level grammar of the specific language
    and identify the underlying logical structure. Use only placeholders like P1(x),
     P2(x,y). The permitted logical symbols are: ∀, ∃, ∧, ∨, ⊕, →, ↔, ¬, ( ),

Example 1: Universal Quantifier & Implication
- (English) NL: Every student who studies hard passes the exam.
- (Other languages)
FOL Skeleton: ∀x (P1(x) ∧ P2(x) → P3(x))

Example 2: Exclusive Disjunction
- (English) NL: A road can be either paved or unpaved.
- (Other languages)
FOL Skeleton: ∀x (P1(x) → (P2(x) ⊕ P3(x)))

Now complete the following task.
Sentence: {nl_input}
FOL Skeleton:
```

Figure 7: Prompt for the Skeleton Generator agent.

```
You are a First-Order Logic master, skilled in assembling expressions from
    multilingual sources. Your task is to fill the placeholders in the 'FOL Skeleton
    ' with predicates derived from the 'Original Sentence' to form a complete and
    syntactically correct FOL expression.

A crucial rule for predicate generation: Regardless of the input language, you must
    create predicates by following these steps: 1. Identify the core concept. 2.
    Translate it into concise English. 3. Format it in UpperCamelCase. For instance,
     from the Chinese 'shuo duo zhong yu yan de ren', the predicate becomes '
    SpeaksMultipleLanguages(x)'.

Your output must ONLY be the final FOL expression.

Task:
Original Sentence: {nl_input}
FOL Skeleton: {logic_skeleton}
Final FOL Expression:
```

Figure 8: Prompt for the Predicate Combiner agent.

lence) metric, we introduced a **Third-Party Auditor** mechanism.

We utilized **Gemini-2.5-Flash**, a model from a completely different provider and architecture, to cross-validate the judgments of our primary Qwen-based SE judge. We randomly sampled 150 instances from the test set and performed a blind audit. The auditor was tasked with the same 3-way classification: `Equivalent`, `Prediction_Better`, or `Truth_Better`.

As shown in Table 7, we observed an overall agreement rate of **74.0%**. Manual inspection of disagreements revealed they primarily occurred in highly ambiguous edge cases where the semantic boundary of the natural language input was open to interpretation (e.g., vague quantification). This high level of cross-model consistency validates that our SE metric is robust and reliable.

F End-to-End Walkthrough of the MA-FOL Pipeline

To make our proposed generate-then-refine process concrete, this section provides a step-by-step walkthrough of how MA-FOL processes a single sentence from the test set. This example effectively demonstrates the self-correction capability of the Refiner stage and highlights the limitations of traditional evaluation metrics.

Step 1: Input Natural Language. The process begins with the raw Muitilingual NL input sentence.

> **NL (en):** "If a vehicle is a bicycle, then it has two wheels and is human-powered."

```
You are an expert FOL validator and refiner, specializing in standardizing
    expressions from diverse linguistic origins. Your task is to ensure the final
    FOL expression adheres to a high-quality, standardized format.

You will be given an 'Original Sentence', a 'Proposed FOL', and a list of 'High-
    Quality Examples'.

Instructions:
1. Analyze the 'Original Sentence' to understand its abstract logical pattern.
2. Check if this abstract pattern closely and unambiguously matches one of the 'High
    -Quality Examples'.
3. Decision Logic:
    - IF IT MATCHES a high-quality example: DISCARD the 'Proposed FOL' and generate
    a new FOL strictly following the example's template.
    - IF IT DOES NOT MATCH any example: REFINE the 'Proposed FOL' and output it .
4. Your output MUST ONLY be the final, complete FOL expression.

High-Quality Examples of Logical Patterns:
{examples_str}

Now, perform your validation and refinement task.
Original Sentence: {nl_input}
Proposed FOL: {multi_agent_fol}
Final FOL Expression:
```

Figure 9: Prompt for the Dynamic RAG-Refiner agent.

		Gemini-2.5-Flash Auditor	
		Positive (Pass)	*Negative* (Fail)
Qwen Judge	*Positive* (Pass)	**Consistent** (Both Approve)	Disagreement (Qwen only)
	Negative (Fail)	Disagreement (Gemini only)	**Consistent** (Both Reject)
Agreement Rate		**74.0%** (111/150)	

Table 7: Confusion Matrix showing the alignment between our primary Qwen-based Judge and the third-party Gemini-2.5-Flash Auditor. 'Positive' indicates the judge rated the prediction as `Equivalent` or `Prediction_Better`. The substantial agreement rate (74.0%) confirms that the SE metric reflects objective semantic truth rather than model-specific bias.

Step 2: Skeleton Generator Output. The NL sentence is passed to the Skeleton Generator, which extracts the language-agnostic logical structure. At this stage, the agent simplifies the complex subject "a vehicle is a bicycle" into a single placeholder ‘P1(x)‘.

> **Generated Skeleton:** $\forall$x (P1(x) $\rightarrow$ (P2(x) $\wedge$ P3(x)))

Step 3: Predicate Combiner Output (Intermediate FOL). The Predicate Combiner then fills the skeleton. This results in a syntactically correct but semantically suboptimal formula with a compound predicate `Vehicle_is_bicycle(x)`.

> **Intermediate FOL:** $\forall$x (Vehicle_is_bicycle(x) $\rightarrow$ (Has_two_wheels(x) $\wedge$ Human_powered(x)))

Step 4: Dynamic RAG-Refiner Correction and Final Output. The Intermediate FOL is passed to the Dynamic RAG-Refiner. By matching the input NL's pattern against its internal exemplars, the refiner identifies the suboptimal structure. It then regenerates a more precise and correctly structured final FOL.

> **Final Refined FOL:** $\forall$x (Vehicle(x) $\wedge$ Bicycle(x) $\rightarrow$ (HasTwoWheels(x) $\wedge$ HumanPowered(x)))

This step is crucial, as the refiner successfully decomposes the initial predicate into the more accurate logical conjunction `Vehicle(x)` $\wedge$ `Bicycle(x)`, demonstrating the framework's key self-correction capability.

Step 5: Final Evaluation and The Limitation of LE Metric. The final, refined output is then evaluated against the ground-truth FOL.

> **Ground Truth FOL:** $\forall$x (Vehicle(x) $\wedge$ Bicycle(x) $\rightarrow$ HasTwoWheels(x) $\wedge$ HumanPowered(x))

This example highlights the brittleness of existing metrics: despite being logically identical to the ground truth (differing only by redundant parentheses), the prediction received a Logical Equivalence

```
You are an expert in both natural language semantics and formal logic. Your task is
    to perform a three-way comparison between a natural language sentence (NL), a
    ground truth First-Order Logic expression (FOL_1), and a predicted FOL
    expression (FOL_2).
Here is the data:
Natural Language (NL): "{nl_sentence}"
Ground Truth FOL (FOL_1): "{true_fol}"
Predicted FOL (FOL_2): "{predicted_fol}"
Follow these steps to make your judgment:
1. First, determine if FOL_1 and FOL_2 are logically equivalent.
2. If they are NOT equivalent, analyze which one is a more accurate and faithful
    translation of the Natural Language sentence (NL). Consider all nuances,
    quantifiers, and predicates.
You MUST respond with ONLY ONE of the following three keywords, without any other
    text or explanation:
- equivalent: If FOL_1 and FOL_2 are logically equivalent.
- prediction_better: If they are not equivalent, AND FOL_2 is a better
    representation of the NL than FOL_1.
- truth_better : If they are not equivalent, AND FOL_1 remains the better
    representation of the NL.
```

Figure 10: Prompt for the LLM-as-Judge in SE metric evaluation.

(LE) score of only 0.625 due to the metric's sensitivity to minor syntax. Conversely, our proposed SE metric, powered by an LLM-as-Judge, correctly identified the equivalence and awarded a full score of 1.0, demonstrating its superior ability to capture true semantic meaning over strict syntactic matching.