LLM Pipeline for Hierarchical Narrative Classification: A Traceable Approach to Multilingual Narrative Detection

Anonymous submission

Abstract

This paper presents Agora, a configurable and reproducible framework for robust hierarchical multi-label classification using large language models. We address critical reliability challenges in LLM-based classification through a multi-agent ensemble approach with voting consensus and an optional actor-critic self-refinement loop. Our system decomposes the hierarchical classification task into manageable steps via a state-graph pipeline, providing enhanced traceability for multilingual propaganda detection. We evaluate our approach on hierarchical narrative classification benchmarks, demonstrating improved robustness and consistency compared to baseline LLM approaches.

Keywords: hierarchical classification, propaganda detection, narrative analysis, large language models, multilingual NLP

1. Introduction

Consider two statements about international relations: (1) *"Western diplomats met with Ukrainian officials to discuss potential peace negotiations,"* versus (2) *"While Western officials claim to pursue peace, their negotiations appear primarily designed to serve their own geopolitical interests rather than address regional security concerns."* While the first presents factual reporting, the second exemplifies a narrative frame: *Discrediting the West: Diplomacy*. This narrative subtly casts doubt on Western intentions through framing choices—questioning motives ("claim to pursue"), suggesting ulterior purposes ("primarily designed to serve their own interests"), and implying disregard for legitimate concerns. Detecting such narratives requires hierarchical classification: identifying the top-level frame (*Discrediting the West*), the specific narrative dimension (*Diplomacy*), and potentially nested sub-narratives (e.g., *questioning motives*, *serving own interests*). Such multi-layered structures are pervasive in online discourse, where persuasive messaging often combines several rhetorical strategies simultaneously to influence public opinion.

Text Classification (TC) is a foundational task in Natural Language Processing (NLP) (Zangari et al., 2024). While traditional approaches often model TC as a single-label problem, real-world texts frequently contain multiple overlapping themes, motivating the use of Multi-Label Classification (MLC) (Hu et al., 2025; Tidake and Sane, 2018). A particularly challenging yet crucial variant is Hierarchical Multi-Label Classification (HMLC), where labels are organized in a predefined hierarchy (e.g., a tree or DAG) (Liu et al., 2023). Narrative detection exemplifies this challenge: narratives are structured frameworks that shape how information is presented and interpreted, often appearing in nested relationships where broad themes (e.g., "climate skepticism") contain more specific sub-narratives (e.g., "scientists are biased," "data is manipulated"). Accurately identifying these nested structures is essential for understanding persuasive communication in misinformation, news framing, and political discourse.

Large Language Models (LLMs) have opened new frontiers for HMLC, enabling powerful zero-shot classification without extensive labeled data. But this power comes with reliability challenges. LLMs are known to be stochastic, producing different outputs for the same input, and often exhibit low instruction fidelity, failing to consistently adhere to complex hierarchical constraints or output formats (Qin et al., 2024). These limitations hinder their deployment in high-stakes applications where robustness and verifiable reasoning are paramount.

To address these shortcomings, we introduce **Agora**, a multi-agent ensemble framework that significantly improves the robustness and accuracy of LLMs on complex HMLC tasks. By aggregating parallel classifications from multiple LLM agents via voting, Agora mitigates the inherent stochasticity of LLMs and produces more reliable results than a standard single-agent approach. We evaluate our framework on SemEval-2025 Task 10 for multilingual narrative classification (Piskorski et al., 2025), demonstrating state-of-the-art performance across five languages (Bulgarian, English, Hindi, Portuguese, Russian). Our main contributions are:

- A multi-agent ensemble framework Agora that uses consensus-based voting to improve reliability for HMLC, achieving first place in Hindi and top-3 rankings in four other languages on post-challenge SemEval-2025 Task 10.
- A comprehensive evaluation showing that consensus-based ensembles outperform both naive baselines and actor-critic self-refinement approaches, with an 11% F1-Samples improvement over single-agent classification.
- An investigation of voting strategies (union, intersection, majority vote) showing that intersec-

tion voting provides the best precision-recall balance for hierarchical narrative detection.

2. Related Work

The core challenge in Multi-Label Classification (MLC) has historically been the effective modeling of inter-label dependencies. Early methods sought to adapt single-label algorithms to this task. Approaches ranged from Binary Relevance, which simplifies the problem by training an independent classifier for each label but ignores correlations, to Classifier Chains, which attempt to capture dependencies by feeding the predictions of one classifier as features to the next in a sequence (Zhang et al., 2018; Read et al., 2011). While foundational, these methods were often superseded by deep learning models, particularly Transformers, which could learn complex label correlations implicitly from large datasets (Devlin et al., 2019).

Building on these advances, research into Hierarchical Multi-Label Classification (HMLC) focused on creating architectures that explicitly leverage the label taxonomy. A dominant paradigm involved coupling a Transformer-based text encoder with a Graph Neural Network (GNN) that operates on the label graph, allowing the model to learn hierarchy-aware representations and enforce structural consistency (Zhou et al., 2020; Xu et al., 2021). But the advent of Large Language Models (LLMs) has fundamentally shifted the landscape, enabling powerful zero-shot HMLC capabilities that bypass the need for complex, task-specific architectures and extensive supervised training (Wang et al., 2023).

The recent SemEval-2025 Task 10 on narrative detection (Piskorski et al., 2025) exemplifies this modern, LLM-driven approach. State-of-the-art systems effectively use techniques like retrieval-augmented generation and specialized prompting to classify texts within the task's two-level hierarchy (Singh et al., 2025; Younus and Qureshi, 2025). Prior work has also introduced modular "agentic" frameworks that decompose the task into parallel binary decisions (Eljadiri and Nurbakova, 2025). Yet, these pioneering systems share a fundamental vulnerability: their reliance on a single, non-deterministic generative pass, which exposes them to the inherent stochasticity of LLMs.

While ensembling is a well-established technique for improving robustness in supervised models (Jurkiewicz et al., 2020), its systematic application to mitigate the unreliability of zero-shot LLMs in HMLC remains underexplored. This represents a critical research gap: the stochasticity of individual LLM calls stands in stark contrast to the deterministic requirements of hierarchical classification systems, yet few works have investigated consensus-driven approaches to address this mismatch. Our work directly addresses this gap, proposing a multi-agent ensemble framework to produce stable, consensus-driven classifications that can reliably handle the demanding constraints of HMLC.

3. System Architecture

We address the following problem: given a multilingual news article, assign multiple labels from a two-level taxonomy of narratives and subnarratives.

A promising recent approach to HMLC, which we refer to as **H3Prompting**, was introduced by Singh et al. (2025). This method decomposes the hierarchical classification task into a sequence of zero-shot LLM queries, where an initial prompt determines the top-level category, and subsequent prompts classify second-level narratives conditioned on the first-level prediction. In the spirit of reproducible research, we first aimed to replicate this method. However, the lack of publicly available implementation details prevented a direct reproduction, motivating our development of an open and configurable framework to systematically investigate zero-shot HMLC strategies.

Our investigation into a robust zero-shot HMLC framework followed an iterative design process. First, we propose a traceable self-refining pipeline. Second, upon identifying its limitations, we present our more robust multi-agent ensemble approach.

3.1. Traceable Actor-Critic Pipeline

Our **Actor-Critic Pipeline** aims to improve the reliability and transparency of a single agent's output. In zero-shot classification context, simple label outputs lack traceability and auditability, and are susceptible to unverified hallucinations. To overcome this, we propose a multi-stage pipeline reframing the task from simple classification to evidence-grounded claim generation and validation.

Our implementation uses the LangGraph framework to structure the process as a cyclical state graph (LangChain AI, 2024a). The pipeline configuration is managed by a `ConfigurableGraphBuilder`, allowing components like validation to be enabled or disabled and different LLMs to be assigned to specific nodes via a YAML file. For the Actor-Critic experiments, we use Gemini 2.5 Flash as the underlying LLM for both the Actor and Critic agents. The workflow proceeds through several stages (detailed below and visualized in Appendix Figure 2).

Category Classification. An initial node determines the document's domain (e.g. Climate Change (CC) or Ukraine-Russia-War (URW)) to focus subsequent stages on the relevant subset of the narrative taxonomy.

The Narrative Actor (Claim Generation). The "Actor" is an LLM agent prompted to act as an expert analyst. For each narrative it identifies, it must generate a structured JSON object containing the `narrative_name`, a verbatim `evidence_quote` from the text, and reasoning that connects the two.

The Narrative Critic (Claim Validation). The Actor's output is passed to the "Critic," a separate LLM agent with a distinct, skeptical persona. The Critic validates the Actor's claims against strict criteria (evidence accuracy, relevance, completeness). If the Critic finds flaws, it generates structured feedback, and the graph's conditional logic routes the state back to the Actor for a retry, incorporating the feedback into a refinement prompt.

Sub-narrative Actor and Critic. Once narratives are set, the process repeats hierarchically. A Sub-narrative Actor generates claims for each parent narrative, which are then validated by a Sub-narrative Critic, also with a self-correction loop.

This evidence-grounded architecture offers significant advantages in traceability and auditability, as every prediction is grounded in a textual `evidence_quote`, a process we enhanced using LangSmith for detailed debugging and prompt refinement (LangChain AI, 2024b). However, while this pipeline showed promise, our experiments revealed that it introduced its own form of instability (see Table 2). The process was highly sensitive to the quality of the Critic's feedback, which, being LLM-generated, was itself prone to stochasticity. A flawed critique could send the Actor down an unproductive refinement path, sometimes degrading performance. This reliance on a single "critic" highlighted a fundamental bottleneck and motivated our shift towards a consensus mechanism.

3.2. Prompt Engineering Strategy

A critical component of both the Actor-Critic and Agora frameworks is the systematic design of prompts. Our prompting strategy incorporates several key principles:

1. Structured Output Format. All classification prompts enforce structured JSON output with required fields (`narrative_name`, `evidence_quote`, and `reasoning`). This ensures that every classification decision is explicitly grounded in textual evidence, enabling traceability and facilitating automated validation. For example, the narrative classification prompt specifies:

> *"Your entire response MUST be a single JSON object. ALL fields in the schema below are REQUIRED. Do not omit any fields."*

This strict formatting requirement forces the model to provide explicit justifications, preventing opaque label predictions.

2. Chain-of-Thought Reasoning. Prompts explicitly instruct the model to perform step-by-step internal reasoning before producing the final classification. This approach, inspired by chain-of-thought prompting (Wei et al., 2022), encourages the model to identify key phrases, evaluate each potential label *vs.* the evidence, and formulate explicit justifications. The narrative classification prompt includes:

> *"Step 1: Chain of Thought (Internal Reasoning). First, think step-by-step to analyze the provided text. Identify key phrases, arguments, and themes. For each potential narrative from the list, consider if it applies. Find a specific, direct quote from the text that serves as the strongest evidence for each narrative you believe is present."*

This two-step process (reasoning, then formatting) reduces impulsive classifications and improves output quality.

3. Hierarchical Prompt Design We use distinct prompts at each level of the hierarchy:

- **Category Classification**: A strict topical classifier that determines text domain (e.g. Climate Change (CC) or the Ukraine-Russia War (URW)). The prompt enforces strict labeling:

  > *"Output EXACTLY one label token enclosed in square brackets on the next line: [URW], [CC], or [Other]."*

- **Narrative Classification**: Given the domain, the model selects from domain-specific narratives, with each narrative accompanied by its definition, examples, and classification instructions according to the template:

  > *"- {Category}: {Narrative Name}*
  > *Definition: {Definition text}*
  > *Example: {Example text}*
  > *Instruction: {Instruction text}"*

- **Sub-narrative Classification**: Given a parent narrative, the model identifies more fine-grained sub-narratives, including an "Other" option for cases where the text supports the parent narrative but does not fit specific sub-narrative definitions. The prompt explicitly instructs:

  > *"Step 2: Check for a Remainder. Are there any other phrases or arguments that support the parent narrative but were NOT used as evidence for the specific subnarratives? Step 3: Add 'Other' if Necessary."*

4. Critic Prompts and Refinement. In the Actor-Critic pipeline, the Critic agent receives a specialized prompt that adopts a "meticulous and skeptical editor" persona. The Critic evaluates classifications against strict criteria: evidence accuracy (verbatim quotes), relevance (direct support for the label), and completeness (no obvious narratives missed). The critic prompt states:

> *"You are a meticulous and skeptical editor. Your task is to evaluate a classification of propaganda narratives applied to a text. You must be extremely strict. The classification is only valid if every narrative is strongly and explicitly supported by the provided evidence from the text."*

When validation fails, a refinement prompt incorporates the Critic's feedback:

> *"You previously analyzed a text, but your analysis had flaws. A meticulous editor has provided the following feedback. Your task is to re-analyze the text, incorporating this feedback to produce a new, corrected classification."*

This feedback loop enables iterative self-correction.

The complete prompts used in our experiments are provided in Appendix A.2.

3.3. Proposed Framework: Agora Multi-Agent Ensemble

To overcome the limitations of a single-critic system and the broader issue of single-agent stochasticity, we developed Agora, a multi-agent ensemble framework. The core principle of Agora is to replace the judgment of a single agent (or a single critic) with the collective "wisdom of the crowd," leveraging the consensus of multiple independent agents to arrive at a more stable and accurate classification.

Importantly, Agora uses the same core prompts as the baseline and Actor-Critic systems (without the Critic validation step). This design choice isolates the effect of the ensemble mechanism itself, allowing us to attribute performance improvements specifically to the consensus-based aggregation rather than to prompt engineering differences. For the Agora experiments, we use GPT-5-nano as the underlying LLM for all agents in the ensemble, with $N = 3$ agents by default.

The framework operates via a fan-out, fan-in process for each level of the hierarchy (e.g., Narrative classification).

Fan-Out (Parallel Classification). Instead of a single LLM call, Agora instantiates N independent agents (where N is a configurable parameter). The same input text and prompt are sent to all N agents, who perform the classification in parallel. This step effectively samples N independent points from the LLM's output distribution for the given task.

Fan-In (Aggregation via Voting). After all N agents return their individual classifications, an Aggregation node consolidates the results using a voting scheme. We implemented and evaluated three distinct aggregation strategies:

- **Union:** The final label set includes any label proposed by at least one agent. This high-recall strategy is useful for identifying all potential labels.
- **Intersection:** The final label set includes only those labels that all agents unanimously agreed upon. This high-precision strategy filters out all but the most confident predictions.
- **Majority Vote:** The final label set includes any label proposed by more than half ($> N/2$) of the agents. This strategy provides a robust balance between precision and recall, filtering out stochastic, outlier classifications while retaining a strong consensus.

This ensemble approach directly addresses the single-point-of-failure issue observed in the Actor-Critic pipeline. Instead of relying on one critic's potentially noisy feedback, Agora relies on the statistical robustness of a majority decision, providing a more reliable and conceptually simpler method for improving classification quality.

4. Experimental setup

The **dataset** from the SemEval-2025 Task 10, Subtask 2 (Piskorski et al., 2025) spans five languages Bulgarian (BG), English (EN), Hindi (HI), Portuguese (PT), and Russian (RU). Each article is annotated with one or more narrative labels from a predefined set of 22 top-level narratives and 95 possible sub-narratives. The train dataset contains a total of 3,874 narrative annotations and 3,874 corresponding subnarrative annotations

While we focus our evaluation mainly on the English subset, our analysis reveals several key characteristics that motivate our architectural choices:

1. Multi-label nature: A majority of documents (54.0%) are assigned more than one narrative, with an average of 2.28 2.28 narratives and 2.28 subnarrative labels per document. This necessitates a multi-label modeling approach.
2. Severe class imbalance: As illustrated in Appendix Figure 3, the 22 narrative labels follow a long-tailed distribution. The most frequent narrative (*URW: Discrediting Ukraine*) appears 65 times more often than the least frequent one (*CC: Green policies are geopolitical instruments*). This sparsity poses a significant challenge for traditional supervised models,

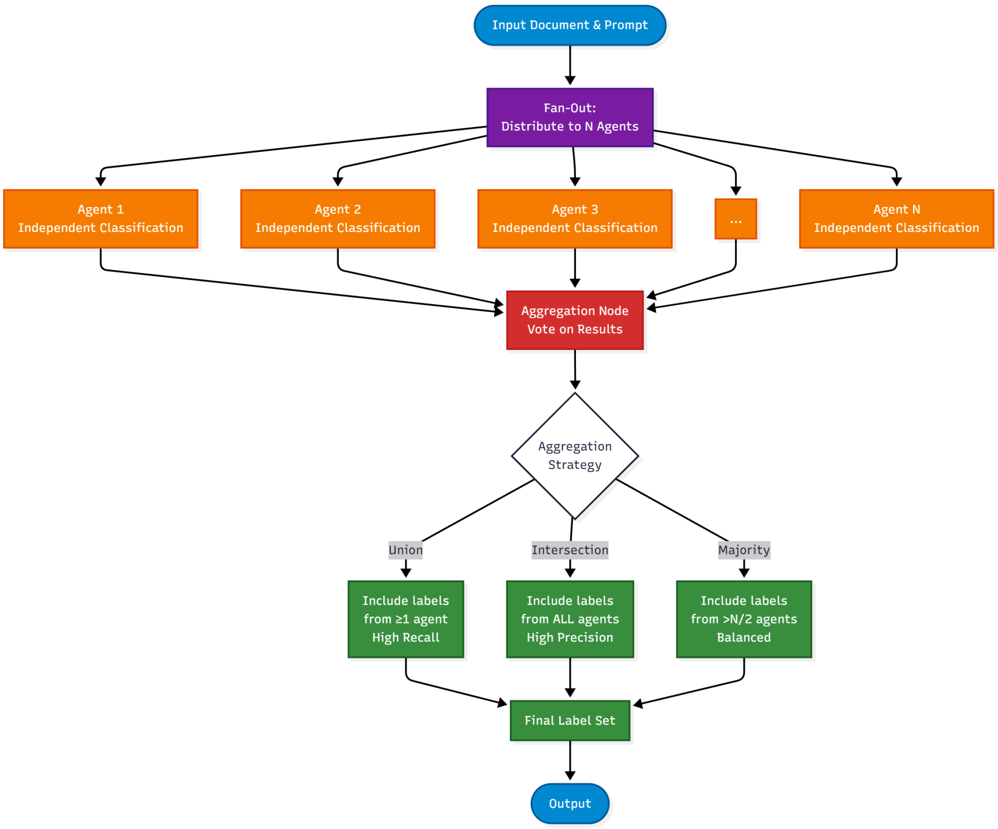
Input Document & Prompt
Fan-Out:
Distribute to N Agents
Agent 1
Independent Classification
Agent 2
Independent Classification
Agent 3
Independent Classification
...
Agent N
Independent Classification
Aggregation Node
Vote on Results
Aggregation Strategy
Union
Intersection
Majority
Include labels from ≥1 agent High Recall
Include labels from ALL agents High Precision
Include labels from >N/2 agents Balanced
Final Label Set
Output

Figure 1: Agora multi-agent ensemble architecture. The classification task is distributed to N independent agents in parallel (Fan-Out), aggregates their results using a voting mechanism (Fan-In), and produces a final label set based on the chosen aggregation strategy (Union, Intersection, or Majority Vote).

which risk overfitting on the few "head" classes and failing to generalize to the many rare but meaningful "tail" classes. This characteristic strongly motivates our adoption of a zero-shot paradigm, which does not depend on label frequencies in a training set.

3. Cross-lingual distribution: The language-specific statistics reveal variation in annotation density: BG (401 docs, 855 annotations, avg 2.13 labels/doc), EN (399 docs, 875 annotations, avg 2.19 labels/doc), HI (366 docs, 655 annotations, avg 1.79 labels/doc), PT (400 docs, 1,217 annotations, avg 3.04 labels/doc), and RU (133 docs, 272 annotations, avg 2.05 labels/doc). Portuguese exhibits notably higher annotation density, while Hindi shows lower average labels per document.

4. Text lenth: Document lengths vary considerably, ranging from 45 to 4,422 words, with an average of 404 words and a median of 371 words. This variation requires robust handling of both short and long-form content.

Due to the multi-label nature and severe class imbalance inherent in the dataset, we decided to adopt a zero-shot learning approach. This paradigm avoids dependency on large per-class training counts and is well-suited to handle the long-tail distribution of narratives, making it ideal for this challenging classification task.

All experiments were conducted on a machine with the following specifications: Processor: Intel Core 9 Ultra; **GPU:** NVIDIA RTX 4070 (8GB VRAM); **Memory:** 32GB RAM; **Storage:** 1TB SSD.

We primarily evaluate the performance of our models on the English test set, providing detailed analysis and interpretation of the results. However, to assess the multilingual capabilities of our approaches, we also conducted experiments on additional languages to validate the generalizability of our methods across different linguistic contexts.

5. Results

In this section, we report our empirical results. We first provide a detailed architectural comparison to understand the source of improvements, analyze voting strategies, and then validate our approach through competitive performance in the official SemEval-2025 Task 10 shared task.

5.1. Architectural Comparison and Ablation Study

To understand the effectiveness of our approach, we conducted a comparison on the English dataset between our final Agora system, an intermediate Actor-Critic pipeline, and a Naive Baseline. The latter represents the H3Prompting approach described in Section 3: a single agent per level, applying one LLM call per hierarchical step without refinement. The results are presented in Table 1.

This ablation study reveals a clear performance pathway. The Naive Baseline achieves an F1-Samples score of 0.382. The Actor-Critic Pipeline provides a significant improvement of +0.035, demonstrating that a self-refinement loop can effectively correct errors and improve classification quality. However, our proposed Agora framework delivers an even greater performance gain. It achieves the highest F1-Samples score of 0.424, a +0.042 point improvement (+11.0%) over the Naive Baseline. This result confirms our central hypothesis: while single-agent refinement is beneficial, the consensus-based approach of a multi-agent ensemble is a more robust and superior method for enhancing LLM reliability.

5.2. Agora's Voting Strategies

To isolate the effect of the ensemble itself, we compared a single-agent Agora configuration ("Vanilla") against the different multi-agent voting strategies. Table 1 shows that the choice of aggregation method is critical to the ensemble's success.

This analysis provides a crucial insight: simply combining agent outputs is not enough. The Union strategy, which naively accepts all proposed labels, actually performs worse than a single agent by aggregating noise. In contrast, the consensus-based mechanisms deliver clear benefits. Both Majority Vote (+0.013) and Intersection (+0.035) significantly outperform the single agent. The Intersection strategy, requiring unanimous agreement, proves most effective for this task, successfully filtering out stochastic, low-confidence predictions to achieve the best overall performance. This demonstrates that the core advantage of Agora lies in its ability to systematically reduce noise through consensus.

5.3. Actor-Critic Validation Ablation: More Complexity Does Not Equal Better Performance

To further understand the refinement paradigm, we conducted an ablation study of the Actor-Critic pipeline across the multilingual development set (178 documents). We evaluated three validation configurations: no validation (baseline single-pass), narrative-level validation only, and subnarrative-level validation only. Table 2 presents the results.

The ablation study reveals a counter-intuitive but critical finding: **adding validation mechanisms generally degrades performance rather than improving it**. On average, the no-validation baseline achieves the highest F1 Coarse (0.6019) and competitive F1 Sample (0.4077), while both validation strategies show negative average deltas for F1 Coarse (-0.0100 for narrative, -0.0261 for subnarrative). Russian is the only language where narrative validation consistently improves both metrics (+0.0122, +0.0032), while Portuguese experiences the most dramatic degradation (-0.0585, -0.0370 with subnarrative validation). This pattern suggests that critique agents systematically over-correct the base model's predictions, introducing noise rather than filtering it. The finding directly motivates our shift to Agora's consensus-based approach, which achieves improvements through diversity and aggregation rather than iterative refinement that risks over-correction.

5.4. Competitive Performance

The main validation of our approaches comes from their performance compared to the official SemEval-2025 Task 10 results. Table 3 presents a comprehensive comparison across all five languages. We compare our solutions with the winning teams (GATENLP for EN, PT, RU, PATeam for BG, QUST for HI). Note that we were not able to reproduce the results of the main winning system (GATENLP).

The results reveal several critical insights about the two architectures. On average, Agora achieves a +0.028 improvement in F1 Samples over the Actor-Critic pipeline (0.446 vs 0.418), while maintaining comparable F1 Macro performance (0.585 vs 0.584). More importantly, Agora's improved average ranking from 3.2 to 2.4 demonstrates superior competitive performance.

Hindi shows dramatic improvement. The most striking result is Hindi, where Agora's Full Intersection configuration achieves 0.581 F1 Samples (+0.146 improvement, +33.6%) and 0.673 F1 Macro, elevating the ranking from 3rd to 1st place. This substantial gain suggests that consensus-based ensembling is particularly effective for languages where single-agent predictions exhibit higher variance.

Configuration selection is language-dependent. The optimal Agora configuration varies significantly across languages: Bulgarian benefits from Narrative Intersection, English and Hindi achieve best results with Full Intersection, while Portuguese and Russian perform optimally with Narrative Union. This pattern indicates that different languages and their associated narrative distributions require different consensus strategies,

Table 1: Main Performance Comparison on the English Dataset (F1-Samples).

System Configuration	F1-Samples Score	Improv. vs. Baseline	Improv. vs. Agora Single Agent
Naive Baseline (Single-Pass)	0.382	—	-0.007
Actor-Critic Pipeline	0.417	+0.035	+0.028
Agora Single Agent (Vanilla)	0.389	+0.007	—
Agora 3-Agent Union	0.375	-0.007	-0.014
Agora 3-Agent Majority Vote	0.402	+0.026	+0.013
Agora 3-Agent Intersection	**0.424**	**+0.042**	**+0.035**

Table 2: Ablation study of Actor-Critic pipeline validation across multilingual dev set (178 documents)

Language	Validation	F1 Coarse	F1 Sample	Δ F1 Coarse	Δ F1 Sample
BG	None	0.5799	0.4104	—	—
	Narrative	0.5721	0.4049	-0.0078	-0.0055
	Subnarrative	0.5521	0.4211	-0.0278	+0.0107
EN	None	0.5132	0.3815	—	—
	Narrative	0.5241	0.3675	+0.0109	-0.0140
	Subnarrative	0.5319	0.4166	+0.0187	+0.0351
HI	None	0.4715	0.2976	—	—
	Narrative	0.4217	0.3082	-0.0498	+0.0106
	Subnarrative	0.4354	0.2687	-0.0361	-0.0289
PT	None	0.7489	0.4737	—	—
	Narrative	0.7335	0.4681	-0.0154	-0.0056
	Subnarrative	0.6904	0.4367	-0.0585	-0.0370
RU	None	0.6958	0.4754	—	—
	Narrative	0.7080	0.4786	+0.0122	+0.0032
	Subnarrative	0.6693	0.5078	-0.0265	+0.0324
Average	None	**0.6019**	**0.4077**	—	—
	Narrative	0.5919	0.4055	-0.0100	-0.0022
	Subnarrative	0.5758	0.4102	-0.0261	+0.0025

with Full Intersection particularly effective for languages like Hindi with high prediction variance.

Mixed performance on English and Portuguese. Interestingly, Agora shows slight degradation on English (-0.009) and more substantial decrease on Portuguese (-0.048) despite maintaining competitive rankings. This suggests that for languages with certain characteristics, the Actor-Critic's iterative refinement approach may be more effective than consensus-based voting, particularly when the base model predictions are already highly accurate.

Consistent ranking improvements validate ensemble approach. Despite mixed F1 Samples results, Agora achieves better or equal rankings in all languages, most notably improving Bulgarian (5th→4th) and Russian (4th→3rd) beyond the dramatic Hindi improvement (3rd→1st). This validates that consensus-based ensembling provides more robust competitive performance across diverse multilingual settings.

These top-ranked results confirm that Agora's consensus mechanism achieves superior average performance (+0.028 F1 Samples) compared to Actor-Critic approach and has competitive standings (2.4 vs 3.2 average rank), particularly excelling in languages with high prediction variance (Hindi).

6. Discussion

The central finding of our work is that consensus through ensembling is a more effective strategy for improving LLM reliability than single-agent self-correction. Our experiments show a clear progression: a Naive Baseline suffers from the model's inherent stochasticity; an Actor-Critic pipeline attempts to fix this with a single "referee" but proves inconsistent, as the referee itself is an unreliable LLM. The Agora framework succeeds because it replaces this fragile, single point of correction with the statistical power of a majority vote.

By sampling multiple outputs from the LLM's distribution and aggregating them, Agora effectively averages out random hallucinations and individual agent errors. The ablation study on voting strategies (Table 1) confirms this: the noisy Union strategy underperforms, while consensus-based methods like Majority Vote and Intersection successfully filter out outlier predictions, leading to higher-quality classifications. This architectural choice directly addresses the core problem of LLM stochasticity and is the primary reason for Agora's top-ranked performance on the SemEval-2025 dataset.

Furthermore, our initial experiments with the Actor-Critic pipeline serve as a crucial cautionary tale. The finding that the validation step often de-

Table 3: Comprehensive comparison of Actor-Critic and Agora architectures in SemEval-2025 Task 10 across all languages and vs. the winning run from the official leaderboard of SemEval Task 10. We were not able to reproduce the results of the latter. F1 Samples is the primary evaluation metric, with F1 Macro Coarse provided for reference. In parenthesis, a post-challenge ranking is given. Agora configurations shown represent the best-performing variant for each language.

Language	**Actor-Critic (Gemini 2.5 Flash)**		**Agora (GPT-5-nano)**			**Best Official**	
	F1 Samples	**F1 Macro**	**F1 Samples**	**F1 Macro**	**Config**	**F1 Samples**	**F1 Macro**
BG	0.381 (5)	0.590	0.403 (4)	0.575	Narr. Inter.	0.460	0.631
EN	0.433 (2)	0.547	0.424 (2)	0.518	Full Inter.	0.438	0.590
HI	0.435 (3)	0.515	**0.581** (**1**)	**0.673**	Full Inter.	0.535	0.569
PT	0.433 (2)	0.679	0.385 (2)	0.602	Narr. Union	0.480	0.664
RU	0.410 (4)	0.589	0.437 (3)	0.556	Narr. Union	0.518	0.709
Average	0.418 (3.2)	0.584	0.446 (**2.4**)	0.585	—	0.486	0.6326

graded performance suggests that adding architectural complexity, even when well-motivated, does not guarantee improvement. In multi-step reasoning chains, errors can accumulate, and a flawed "critic" can be more harmful than no critic at all. This reinforces the elegance of the Agora approach: it improves robustness through a conceptually simple and statistically grounded method rather than a complex and potentially brittle reasoning loop.

7. Limitations and Future Work

Our study, while demonstrating the effectiveness of the Agora framework, is subject to several **limitations** that offer avenues for future work.

Model and API Dependency: Our zero-shot framework relies on proprietary LLMs (e.g., gpt-5-nano). This limits full reproducibility, as performance is tied to specific, closed-source model versions. Future work should explore the effectiveness of this ensemble approach with open-source models to ensure broader accessibility and control.

Prompt Sensitivity: The performance of any zero-shot system is highly sensitive to prompt engineering. While we standardized prompts across configurations, it is possible that the specific phrasing influenced the degree of stochasticity observed and the effectiveness of the voting mechanism.

Computational Cost: Deploying a multi-agent ensemble incurs a direct multiplication of inference cost and latency compared to a single-agent system. In our $N = 3$ configuration, the cost is roughly triple that of the baseline. While the performance gains justified this trade-off in a competitive setting, a critical area for future research is exploring cost-reduction techniques, such as using smaller, distilled models for some agents or implementing more sophisticated routing where an ensemble is only triggered for high-uncertainty cases.

Several promising directions emerge from this work. First, exploring weighted majority voting or dynamic ensemble triggering for high-uncertainty cases could optimize the cost-performance trade-off. Second, applying Agora to other hierarchical classification domains beyond propaganda detection would validate its generalizability. Finally, extending the framework to fine-tuned models on task-specific data, while addressing the challenge of preserving cultural nuances in multilingual settings, represents an important avenue for further performance improvements.

8. Conclusion

We addressed a Hierarchical Multi-Label Classification challenge using LLMs by introducing Agora, a multi-agent ensemble framework that leverages consensus-based voting to transform unreliable single-agent classification into a robust decision process. Our experiments revealed three key findings: (1) naive single-pass LLM baselines are outperformed by sophisticated architectures, (2) Actor-Critic self-refinement can introduce noise and degrade performance due to critic unreliability, and (3) Agora with 3-agent voting delivers substantial performance gains across all languages, achieving the best results in Hindi and competitive top-tier performance overall. This validates our main claim: multi-agent ensembling through consensus is a practical and effective method for building robust zero-shot classification systems. As NLP increasingly relies on powerful but imperfect LLMs, frameworks prioritizing reliability through consensus will be essential for trustworthy and deployable solutions.

Acknowledgements

We would like to thank the SemEval-2025 organizers for providing the challenging Task 10 on multilingual propaganda narrative detection, which served as an excellent testbed for our research.

A. Appendix

A.1. Architecture Diagrams

This section provides detailed visual representations of the Actor-Critic and Agora architectures described in Section 3.

A.1.1. Actor-Critic Pipeline Architecture

Figure 2 illustrates the complete Actor-Critic pipeline workflow, showing the hierarchical stages and feedback loops described in Section 3.2.

A.2. Prompt Templates

This section provides the complete prompt templates used in our experiments to ensure full reproducibility. All prompts are designed to enforce structured JSON output with evidence grounding.

A.2.1. Category Classification Prompt

You are a strict topical classifier. Decide whether the given text is primarily about URW (Ukraine-Russia War topics) or CC (Climate Change topics).

Rules (follow exactly):

- First find the elements that indicate the topic, and reason through them step by step.
- Output EXACTLY one label token enclosed in square brackets on the next line: `[URW]`, `[CC]`, or `[Other]`.

Classification guidance:

- Use `[URW]` for topics clearly about the Russia-Ukraine conflict.
- Use `[CC]` for topics clearly about climate change.
- Use `[Other]` if neither topic is the primary focus.

Example outputs: `EVIDENCE: short justification` followed by `[URW]`, `[CC]`, or `[Other]`.

A.2.2. Narrative Classification Prompt (Template)

You are an expert propaganda narrative analyst. Your task is to analyze the given text and identify which specific propaganda narratives are present.

AVAILABLE NARRATIVES:

{Category}: {Narrative Name}

Definition: {Definition text}

Example: {Example text}

Instruction: {Instruction text}

INSTRUCTIONS:

Step 1: Chain of Thought (Internal Reasoning). Think step-by-step to analyze the text. Identify key phrases and themes. For each potential narrative, find a specific, direct quote that serves as evidence. Formulate reasoning for why that quote supports the narrative.

Step 2: Format the Final Output. Provide a single, valid JSON object with the following schema: `{"narratives": [{"narrative_name": "string", "evidence_quote": "string", "reasoning": "string"}]}`

A.2.3. Sub-narrative Classification Prompt (Template)

This text is known to contain the narrative: {Parent Narrative}. Your task is to identify which specific sub-narratives are present.

AVAILABLE SUBNARRATIVES: Each sub-narrative includes its definition, example, and classification instruction. An "Other" option is provided for cases where the text supports the parent narrative but does not fit specific sub-narrative definitions.

INSTRUCTIONS:

Step 1: Chain of Thought. Analyze the text step-by-step, looking for specific themes matching sub-narrative definitions. Find direct quotes as evidence.

Step 2: Check for Remainder. Re-read to identify phrases supporting the parent narrative not covered by specific sub-narratives.

Step 3: Add "Other" if Necessary. If uncovered evidence exists, include `{Parent Narrative}: Other`.

Step 4: Format Output. Provide JSON with schema: `{"narratives": [{"subnarrative_name": "string", "evidence_quote": "string", "reasoning": "string"}]}`

A.2.4. Critic Validation Prompt (Narrative Level)

You are a meticulous and skeptical editor. Your task is to evaluate a classification of narratives applied to a text. You must be extremely strict. The classification is only valid if every narrative is strongly and explicitly supported by the provided evidence from the text.

EVALUATION CRITERIA (Apply Strictly):

1. **Evidence Accuracy:** Is the `evidence_quote` an exact, verbatim quote from the original text (allowing for minor formatting differences)?
2. **Relevance of Evidence:** Does the `evidence_quote` DIRECTLY support the `narrative_name`? The connection must not be a stretch.

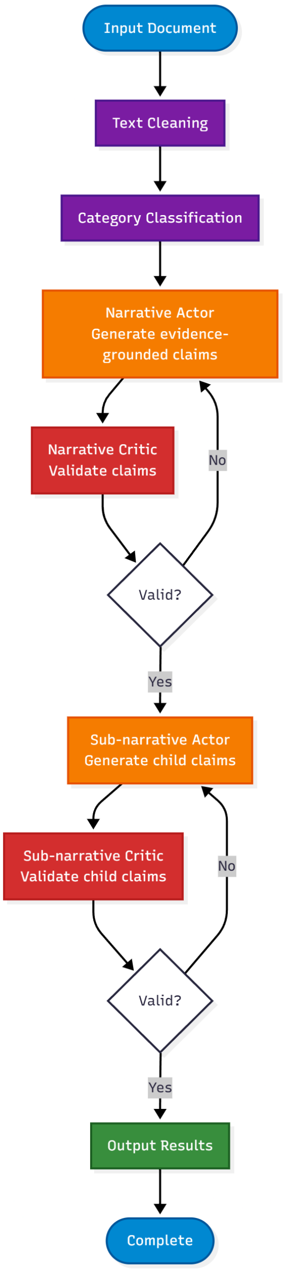

Figure 2: Actor-Critic validation pipeline architecture. The system decomposes HMLC into hierarchical stages where a Narrative Actor generates evidence-grounded claims, which are then validated by a Narrative Critic. Feedback loops enable iterative refinement before progressing to sub-narrative classification.

3. **Completeness:** Does the analysis miss any obvious, high-confidence narratives clearly present in the text?

Output: Provide evaluation as a single, valid JSON object.

A.2.5. Refinement Prompt (After Critic Feedback)

You previously analyzed a text, but your analysis had flaws. A meticulous editor has provided the following feedback. Your task is to re-analyze the text, incorporating this feedback to produce a new, corrected classification.

EDITOR'S FEEDBACK TO CORRECT: {Critic's feedback text}

ORIGINAL TASK AND DEFINITIONS: {Original classification prompt is repeated here}

A.2.6. Text Cleaning Prompt

You are a precise text cleaner. Your job is to clean raw text scraped from the web by removing UI noise and boilerplate while preserving the article's content.

STRICT RULES:

- **REMOVE:** navigation menus, cookie banners, sign-up banners, button labels (e.g., ‘Accept’, ‘Subscribe’, ‘Read more’), share widgets, headers/footers, unrelated CTAs, legal disclaimers, pagination artifacts, and unrelated links.
- **KEEP:** the main article or post content only. Preserve sentence order, punctuation, and language.
- **DO NOT** paraphrase or summarize. Do not add or remove meaning.
- **OUTPUT:** Return ONLY the cleaned text with no additional commentary, headings, or JSON.

A.2.7. Implementation Notes

All prompts are implemented as Python functions that dynamically populate templates with task-specific information (e.g., available narratives, definitions, parent narrative context). The category and narrative definitions are loaded from CSV files containing the task taxonomy. This modular design allows for easy adaptation to other hierarchical multilabel classification tasks. These prompts were specifically developed and evaluated on SemEval-2025 Task 10 for multilingual narrative classification (Piskorski et al., 2025), demonstrating the generalizability of the approach across multiple languages and narrative types.

A.3. Dataset Visualization

A.3.1. Narrative Distribution Visualization

Figure 3 illustrates the severe class imbalance in the SemEval-2025 Task 10 dataset, showing the long-tailed distribution of narrative labels discussed in Section 2.1.

References

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

Mohamed Nour Eljadiri and Diana Nurbakova. 2025. Team INSALyon2 at SemEval-2025 task 10: A zero-shot agentic approach to text classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 965–980, Vienna, Austria. Association for Computational Linguistics.

W. Hu, Q. Fan, H. Yan, X. Xu, S. Huang, and K. Zhang. 2025. A survey of multi-label text classification under few-shot scenarios. *Applied Sciences*, 15:8872.

Dawid Jurkiewicz, Łukasz Borchmann, Izabela Kosmala, and Filip Graliński. 2020. ApplicaAI at SemEval-2020 task 11: On RoBERTa-CRF, span CLS and whether self-training helps them. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1415–1424, Barcelona (online). International Committee for Computational Linguistics.

LangChain AI. 2024a. Langgraph: Build resilient language agents as graphs. Accessed: 2025-09-19.

LangChain AI. 2024b. Langsmith: Developer platform for building reliable llm applications. Accessed: 2025-09-19.

Rundong Liu, Wenhan Liang, Weijun Luo, Yuxiang Song, He Zhang, Ruohua Xu, Yunfeng Li, and Ming Liu. 2023. Recent advances in hierarchical multi-label text classification: A survey.

Jakub Piskorski, Tarek Mahmoud, Nikolaos Nikolaidis, Ricardo Campos, Alípio Jorge, Dimitar Dimitrov, Purificação Silvano, Roman Yangarber, Shivam Sharma, Tanmoy Chakraborty, Nuno Ricardo Guimarães, Elisa Sartori, Nicolas Stefanovitch, Zhuohan Xie, Preslav Nakov, and Giovanni Da San Martino. 2025. SemEval-2025 task 10: Multilingual characterization and extraction of narratives from online news. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval 2025)*, Vienna, Austria.

Tianyi Qin, Yaliang Wu, Yaliang Zhang, Weiliang Liu, Jundong Li, and Philip S. Yu. 2024. Infobench: A benchmark for information disorder detection in social media. *IEEE Transactions on Knowledge and Data Engineering*, 36(5):2132–2145.

J. Read, B. Pfahringer, G. Holmes, and E. Frank. 2011. Classifier chains for multi-label classification. *Machine Learning*, 85(3):333–359.

Iknoor Singh, Carolina Scarton, and Kalina Bontcheva. 2025. GateNLP at SemEval-2025 task 10: Hierarchical three-step prompting for multilingual narrative classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 148–154, Vienna, Austria. Association for Computational Linguistics.

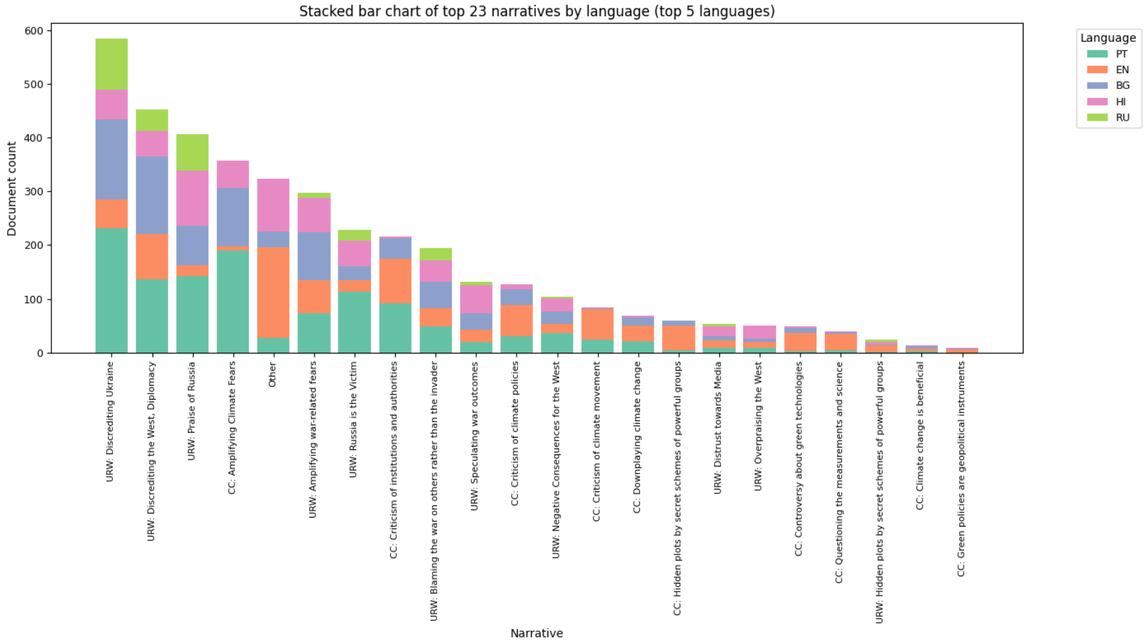

Figure 3: Long-tailed distribution of narrative labels across the dataset. A small number of frequent narratives dominate, while most are rare. This severe imbalance (64.9:1 ratio) justifies the use of zero-shot LLM approaches that do not rely on large per-class training counts.

Vaishali S. Tidake and Shirish S. Sane. 2018. Multi-label classification: a survey. *International Journal of Engineering & Technology*, 7(4.19):1045.

Fengjun Wang, Moran Beladev, Ofri Kleinfeld, Elina Frayerman, Tal Shachar, Eran Fainman, Karen Lastmann Assaraf, Sarai Mizrachi, and Benjamin Wang. 2023. Text2Topic: Multi-label text classification system for efficient topic detection in user generated content with zero-shot capabilities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 93–103, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Linli Xu, Sijie Teng, Ruoyu Zhao, Junliang Guo, Chi Xiao, Deqiang Jiang, and Bo Ren. 2021. Hierarchical multi-label text classification with horizontal and vertical category correlations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2459–2468, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Arjumand Younus and Muhammad Atif Qureshi. 2025. nlptuducd at SemEval-2025 task 10: Narrative classification as a retrieval task through story embeddings. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 1742–1746, Vienna, Austria. Association for Computational Linguistics.

A. Zangari, M. Marcuzzo, M. Rizzo, L. Giudice, A. Albarelli, and A. Gasparetto. 2024. Hierarchical text classification and its foundations: A review of current research. *Electronics*, 13(7):1199.

Min-Ling Zhang, Yu-Kun Li, Xu-Ying Liu, and Xin Geng. 2018. Binary relevance for multi-label learning: an overview. *Frontiers of Computer Science*, 12(2):191–202.

Jie Zhou, Chunping Ma, Dingkun Long, Guangwei Xu, Ning Ding, Haoyu Zhang, Pengjun Xie, and Gongshen Liu. 2020. Hierarchy-aware global model for hierarchical text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1106–1117, Online. Association for Computational Linguistics.