

NARRATOR: Narrative-Aware Routing and Evaluation of Large Language Models

Ahmed Hamdi¹ Jose G. Moreno¹ Antoine Doucet^{2,3} Fabio A. González⁴

¹University of Toulouse, IRIT, Toulouse, France

²L3i, University of La Rochelle, La Rochelle, France

³FRI, University of Ljubljana, Ljubljana, Slovenia

⁴MindLab, Universidad Nacional de Colombia, Bogotá, Colombia

{ahmed.hamdi, jose.moreno}@irit.fr

antoine.doucet@univ-lr.fr

fagonzalezo@unal.edu.co

Abstract

Narrative NLP encompasses a diverse set of competencies such as detecting whether a text contains a narrative, classifying its genre or function, completing its plot, and judging similarity between stories. However, the relationship between model capabilities across these sub-tasks remains poorly understood and no unified benchmark exists to evaluate large language models across these dimensions simultaneously. In this paper, we introduce NARRATOR, a unified benchmark of 96,818 multiple-choice questions derived from six narrative datasets spanning detection, classification, story completion, and similarity. Evaluating 50 open-weight LLMs in a zero-shot setting reveals that LLM performance profiles are highly heterogeneous and no model dominates across all sub-tasks. To exploit this complementarity efficiently, we learn compact narrative competence embeddings of LLMs using a matrix factorisation over a binary correctness matrix, without accessing model weights. A single linear layer on top of these frozen embeddings supports model routing (71.1%), correctness forecasting (65.3%), and benchmark accuracy prediction (mean Kendall's $\tau = 0.72$) at near-zero inference cost.

1 Introduction

Narrative understanding is one of the most demanding frontiers required of NLP systems. It spans a heterogeneous set of sub-tasks: determining whether a text is a narrative (Piper and Bagga, 2025), classifying its genre or discursive function (Kar et al., 2018; Tikhonov et al., 2021; Liu et al., 2022), completing or reasoning about its internal logic (Mostafazadeh et al., 2016), and judging similarity between stories along abstract dimensions of plot, theme, or character arc (Kawada and Holyoak, 2026). Despite sustained research interest, the field remains fragmented: each sub-task is evaluated on its own isolated dataset, domain coverage is narrow, annotation quality varies

widely, and no unified benchmark exists across sub-tasks (Hamilton et al., 2026). As a consequence, it remains unclear how the capabilities of large language models (LLMs) are structured across the full spectrum of narrative competencies. With hundreds of models now available, the heterogeneity of LLMs raises a central practical question: *given a narrative-understanding query, which model should one call?*

The practical difficulty is compounded by the rapid proliferation of open-weight LLMs. With dozens of models available at comparable parameter counts, selecting the right model for a given narrative task is non-trivial. Benchmarking every model exhaustively is increasingly impractical: evaluating a single model across a large narrative benchmark incurs substantial inference costs and must be repeated for every new model or sub-task. Our empirical evaluation of 50 open-weight LLMs on NARRATOR reveals patterns that aggregate scores cannot capture: (1) a 2B model outperforms a 9B model overall, (2) a code-specialised model achieves the highest accuracy on narrative detection despite no explicit training on narrative data, and (3) no single model dominates across all sub-tasks.

We address this challenge from two directions. First, we introduce NARRATOR, a unified benchmark of 96,818 multiple-choice questions derived from six narrative datasets spanning narrative detection, classification, story completion, and narrative similarity. Second, motivated by the observed heterogeneity and complementarity of model profiles, we learn compact *narrative competence embeddings* of LLMs via matrix factorisation over a binary correctness matrix (Zhuang et al., 2025). Rather than accessing model weights or architecture, our framework learns a low-dimensional vector for each model exclusively from its behavioural correctness patterns across questions.

Beyond their practical utility, the learned embeddings offer an analytical window into the structure

of narrative competence across LLMs. Probing experiments show that the geometry of the embedding space encodes theoretically motivated distinctions: sub-tasks cluster by their cognitive demands, annotation-poor datasets such as MND (Liu et al., 2022) (Fleiss $\kappa = 0.24$, near-random model performance) are geometrically isolated from tasks with reliable annotation, and model families form coherent neighborhoods despite differences in size and training regime.

Concretely, this paper addresses two research questions: **(RQ1)** Do LLMs exhibit complementary narrative competence profiles, and are these profiles predictable from model size or architectural family? **(RQ2)** Can a compact vector representation of LLMs, learned from behavioural correctness data alone, support efficient narrative model routing and competence forecasting? Our paper makes the following contributions:

1. **NARRATOR**: a benchmark constructed by systematically converting six heterogeneous narrative datasets (NarraDetect (Piper and Bagga, 2025), MPST (Kar et al., 2018), StoryDB (Tikhonov et al., 2021), MND (Liu et al., 2022), ROCStories/Story Cloze (Mostafazadeh et al., 2016), and SemEval-2026 Task 4 (Hatzel et al., 2026)) into a unified multiple-choice questions (MCQ) format, spanning narrative subtasks including distractor generation, label normalisation, and format validation across detection, classification, completion, and similarity tasks.
2. **A large-scale evaluation of 50 open-weight LLMs** in zero-shot setting, revealing heterogeneous narrative competence profiles: model size is a poor predictor of performance and the oracle routing gain over the single-best model is substantial, motivating sub-task-aware model selection.
3. **A sub-task-aware routing framework** that learns compact narrative competence embeddings of LLMs via matrix factorisation over behavioural correctness data, supporting model routing, correctness forecasting, and benchmark accuracy prediction with a single linear layer and no access to model weights.
4. **Probing experiments** demonstrating that the learned embedding space encodes sub-task structure, model family structure and annotation quality as a geometric signal in the question embedding space.

2 Related Work

We situate NARRATOR within two lines of related work: narrative benchmarking as well as LLM routing and model selection.

2.1 Narrative Benchmarking

Early narrative benchmarks focused on isolated sub-tasks. Mostafazadeh et al. (2016) introduced the ROCStories corpus and the Story Cloze Test, framing narrative understanding as story completion; despite its influence, it covers only everyday fictional narratives and exposes models to minimal linguistic variation. Kar et al. (2018) and Tikhonov et al. (2021) provide genre classification datasets derived from movie synopses and Wikipedia plots, respectively, but they rely on folksonomy tags with no inter-annotator agreement reported. In the political domain, Piskorski et al. (2025) and Fraile-Hernández et al. (2024) target narrative identification in news and social media, with higher annotation quality (Cohen’s $\kappa \approx 0.78$) but narrow domain coverage. Piper and Bagga (2025) introduce the only detection-focused dataset grounded in narrative theory and show that supervised models trained on one domain collapse out-of-domain, while LLMs generalise better. Most recently, Kawada and Holyoak (2026) frame narrative similarity as a structured ranking task. These efforts are complementary but disjoint: no prior work provides a unified, multi-task benchmark enabling direct cross-sub-task comparison of LLM capabilities. NARRATOR fills this gap by aggregating six datasets into a single MCQ format across detection, classification, completion, and similarity.

General-purpose benchmarks such as MMLU (Hendrycks et al., 2020), BIG-Bench (Srivastava et al., 2022), and HELM (Liang et al., 2022) have established the practice of evaluating LLMs across heterogeneous tasks in a zero-shot or few-shot setting. MMLU, in particular, showed that aggregate scores mask large variance across domains, a finding we replicate and sharpen in the narrative domain: sub-task accuracy in NARRATOR ranges from near-chance (MND, $\sim 30\%$ against a 25% baseline) to near-ceiling (ROCStories, $>95\%$ for top models). Myrzakhan et al. (2024) and subsequent community leaderboards have shown that model rankings are highly sensitive to task selection, and that parameter count is a poor predictor of performance on specific domains (Chia et al., 2024).

2.2 LLM routing and model selection

The heterogeneity of LLM capabilities has motivated a line of work on adaptive model selection and routing. [Chen et al. \(2023\)](#) propose a cascade of models ordered by cost, routing queries to cheaper models when confidence is sufficient. [Ong et al. \(2025\)](#) learn a router from human preference data to select between a strong and a weak model, reducing inference cost with minimal quality loss. [Shnitzer et al. \(2023\)](#) train task-specific classifiers to route queries across a fixed pool of models. More recent work has explored routing based on query complexity ([Šakota et al., 2024](#)) or via embedding-based similarity to past queries ([Zeng et al., 2023](#)). These approaches share a common limitation: they require task-specific training and do not generalise across domains or model pools without retraining. Our framework learns a single set of model embeddings from behavioural correctness data without access to model weights or query-level supervision.

The idea of representing models as vectors in a shared latent space was introduced in EmbedLLM ([Zhuang et al., 2025](#)), which showed that matrix factorisation over a binary correctness matrix produces compact LLM embeddings useful for routing, correctness forecasting, and benchmark accuracy prediction across general-domain tasks. Their approach is inspired by collaborative filtering in recommender systems ([Koren et al., 2009](#)), where user and item embeddings are learned jointly to reconstruct an interaction matrix. Prior to EmbedLLM, [Polo et al. \(2024\)](#) showed that small, carefully selected subsets of benchmark questions can accurately predict full benchmark performance, suggesting that correctness patterns carry structured information about model capabilities. [Vivek et al. \(2024\)](#) extend this by learning item response theory (IRT) models over correctness matrices, estimating latent model ability and question difficulty as scalar parameters. Our work specialises this paradigm in narrative NLP, where sub-task structure is theoretically motivated, and provides the first evaluation of behavioural embeddings for domain-specific competence profiling. Specifically, our work differs from EmbedLLM in four ways: (i) a task-aware question encoder with empirical ablation; (ii) application to a theoretically structured domain enabling sub-task geometry analysis; (iii) probing experiments revealing annotation quality as a geometric signal; and (iv) NARRATOR as a primary dataset contribution.

3 Benchmark Construction

3.1 Task Taxonomy

NARRATOR organises narrative NLP into four sub-tasks that reflect the main dimensions along which the field has been structured in the literature ([Hamilton et al., 2026](#)):

- **Narrative Detection:** The task of determining whether a given text contains a narrative, framed as binary classification. Following [Piper and Bagga \(2025\)](#), we adopt a theory-grounded definition of narrativity along three dimensions: Agency, Event Sequencing, and World Building ([Herman, 2017](#)). Negative examples include legal documents, academic articles, and other expository genres.
- **Narrative Classification:** The task of assigning one or more predefined labels to a text already identified as a narrative. This covers genre and tag classification (MPST, StoryDB), narrative function classification (MND), and political narrative identification. Label schemas range from expert-grounded taxonomies to user-generated folksonomy tags, with corresponding variation in annotation reliability.
- **Narrative Similarity:** The task of reasoning about the internal logic of a narrative: how events relate to each other, how stories are structured and what makes them coherent or similar. We operationalise this through story completion (ROCStories / Story Cloze Test) and narrative similarity judgment (SemEval-2026 Task 4).
- **Entity and Role Analysis:** The task of identifying actors in a narrative and determining their role (protagonist, antagonist, or secondary character). This sub-task is absent from the current release due to the lack of qualifying datasets. It is included in the taxonomy as a target for future extension.

3.2 Datasets

Our benchmark aggregates six publicly available datasets, summarised in Table 1. Dataset selection required text availability without API hydration or registration, and compatibility with the MCQ format. For these reasons, two additional datasets, DIPROMATS ([Fraile-Hernández et al.,](#)

Dataset	Sub-task	Choices	Records	Domain	IAA	Avg. chars
StoryDB	Genre classification	4	56,766	Fiction (multilingual→EN)	–	2,447
MPST	Genre classification	4	14,316	Movies	–	5,166
NarraDetect	Narrative detection	2	13,543	Mixed (18 genres)	$\alpha = 0.37$	691
MND	Narrative function	4	5,515	Movies	$\kappa = 0.24$	114
ROCStories	Story completion	2	3,742	Everyday fiction	Human-validated (Mostafazadeh et al., 2016)	185
SemEval-2026 Task-4	Narrative similarity	2	2,936	Movie plots	Annotation-filtered	560
Total			96,818			–

Table 1: Datasets included in NARRATOR, with sub-task, number of answer choices, total records, domain, inter-annotator agreement (IAA) from the original source (– indicates none reported), and average input length in characters. StoryDB covers 42 languages; we retain the English subset only.

2024) and SemEval-25 Task 10 (Piskorski et al., 2025), were excluded due to the unavailability of raw text at the time of construction (tweet IDs only, and registration-gated access, respectively). Moreover, Movie101v2 (Yue et al., 2025) was excluded as it requires video input incompatible with text-based MCQ inference.

Annotation quality. A notable feature of NARRATOR is the wide variation in inter-annotator agreement (IAA) across its constituent datasets. MND reports a Fleiss κ of only 0.24, well below the commonly cited threshold of 0.667 (Landis and Koch, 1977), which the authors themselves acknowledge is descriptive rather than prescriptive, while NarraDetect reports a Krippendorff α of 0.37. MPST and StoryDB report no IAA, relying on user-generated folksonomy tags. This variation is not merely a methodological concern: it constitutes a prior on empirical difficulty for LLMs. MND is retained in NARRATOR not as a diagnostic benchmark but as a calibration case: its near-random model performance and low annotation agreement provide a controlled lower bound that anchors the annotation quality spectrum and validates our hypothesis that IAA is predictive of empirical difficulty (Section 6).

3.3 Data Normalisation

All datasets are converted to a unified MCQ format. Each item contains an id, source, task, input (passage and optional context), question, choices (A–D), answer, and metadata preserving original labels, language, and domain. Samples of this normalisation can be found in Appendix D.

Distractor generation. For datasets with a single correct label (NarraDetect, ROCStories, SemEval-2026 Task 4), the binary format is preserved as a two-choice MCQ. For multi-label classification datasets (MPST, StoryDB), we select the dominant label as the correct answer and sample three distractors from the dataset vocabulary. The dominant

label is defined as the tag among the item’s own annotated tags with the highest corpus-wide frequency. Formally, $\arg \max_{t \in \mathcal{T}_j} \text{freq}(t)$, where $\mathcal{T}_j$ is the item’s tag set and ties are broken by global corpus frequency. For **MPST**, three distractors are sampled uniformly at random from the top-40 vocabulary (tags occurring ≥ 20 times corpus-wide), excluding all of the item’s own tags. For **StoryDB**, distractors are sampled uniformly at random from the top-30 genre vocabulary (tags occurring ≥ 30 times), excluding all of the item’s own tags. For **MND**, the 15 Save the Cat! story beats are mapped to a four-choice MCQ by sampling three distractors uniformly at random from the 12 most frequent beats, excluding the correct beat, with a fallback to the full 15-beat set if fewer than three eligible beats remain. Illustrative transformation examples for MPST and MND are provided in Appendix E. All items follow a single-answer MCQ format: exactly one option is correct, with two options for binary tasks and four for multi-class tasks.

Dataset splits. We perform a random 80%–10%–10% train/validation/test split across the full benchmark pool, yielding 77,454 training, 9,681 validation, and 9,683 test questions. The split is applied globally rather than per dataset. As a result, each source dataset lands at approximately 80% training naturally (Table 2). At the (model, question) pair level, 0.6–0.7% of pairs carry a label of –1 (model failed to produce a valid response) and are excluded from training and evaluation. Format failures are model-specific and show no correlation with question difficulty. The per-model breakdown of –1 label rates is provided in Appendix F. Although our data splits may differ from the original ones to maintain uniform experimental conditions across datasets, this variation does not alter our findings.¹

¹For SemEval-2026 Task 4 specifically, the accuracy figures are not directly comparable to the competition leaderboard due to differences between our splits and those of the task organizers.

Dataset	Train	Val	Test	Total
StoryDB	45,384	5,631	5,751	56,766
MPST	11,396	1,450	1,470	14,316
NarraDetect	10,876	1,377	1,290	13,543
MND	4,459	560	496	5,515
ROCStories	3,003	377	362	3,742
SemEval-2026	2,336	286	314	2,936
Total	77,454	9,681	9,683	96,818

Table 2: Train/validation/test splits across datasets. Counts reflect questions per split.

4 Narrative Competence Embeddings

4.1 Problem Formulation

Let $\mathcal{M} = \{M_1, \dots, M_N\}$ be a set of N LLMs and $\mathcal{Q} = \{q_1, \dots, q_T\}$ be the set of T questions in NARRATOR. We define a binary correctness matrix:

$$\mathbf{Y} \in \{0, 1\}^{N \times T}, Y_{ij} = \mathbf{1}[M_i \text{ answers } q_j \text{ correctly}],$$

where $N = 49$ (excluding TinyLlama) and $T = 96,818$. Each question q_j carries two categorical metadata fields: a dataset source $s_j \in \mathcal{S}$ and a task type $\tau_j \in \mathcal{T} = \{\text{detection, classification, completion, similarity}\}$, both of which are used to condition question representations.

Our goal is to learn two embedding functions: $\phi_m : \mathcal{M} \rightarrow \mathbb{R}^d$ (narrative competence embedding of a model), and $\phi_q : \mathcal{Q} \rightarrow \mathbb{R}^d$ (task-aware difficulty embedding of a question).

4.2 Matrix Factorisation

Building on the EmbedLLM architecture (Zhuang et al., 2025), we extend the question encoder with two narrative-specific conditioning signals, a task-type embedding $\mathbf{e}_\tau$ and question features $\mathbf{f}_j$, while retaining the Hadamard decoder and BCE training objective. The model encoder remains unchanged.

Model encoder. Each model M_i is represented by a trainable embedding vector $\mathbf{v}_m^{(i)} \in \mathbb{R}^d$, initialised from $\mathcal{N}(0, 0.01)$ and optimised via back-propagation. Model weights are never accessed: the encoder is a lookup table over a trainable matrix $\mathbf{E}_m \in \mathbb{R}^{N \times d}$, whose rows constitute the narrative competence embeddings of the N models.

Task-aware question encoder. Each question q_j is encoded by concatenating a frozen sentence transformer embedding (all-mpnet-base-v2, 768-d (Reimers and Gurevych, 2019)), a learned task-type embedding $\mathbf{e}_{\tau_j} \in \mathbb{R}^{d_\tau}$ and scalar question

features $\mathbf{f}_j \in \mathbb{R}^{d_f}$ comprising input length bins, dataset source (one-hot), and estimated difficulty $\hat{d}_j$. The estimated difficulty is defined as $\hat{d}_j = 1 - \frac{1}{N} \sum_i Y_{ij}$, the empirical error rate across all N models in the training split; it is set to 0 for validation and test questions to prevent label leakage. The effect of including $\hat{d}_j$ is analysed in Table 3: it improves routing but degrades forecasting, suggesting that difficulty scores bias the correctness predictor toward the majority class. These three components are concatenated and projected to the shared embedding space:

$$\mathbf{v}_q^{(j)} = h_{\text{proj}}([g_{\text{ST}}(q_j) \parallel \mathbf{e}_{\tau_j} \parallel \mathbf{f}_j]), \quad (1)$$

where $h_{\text{proj}} : \mathbb{R}^{768+d_\tau+d_f} \rightarrow \mathbb{R}^d$ is a single linear layer and $\parallel$ denotes concatenation. This design allows the question encoder to capture both semantic content (via g_{ST}) and structural properties (task type, source, and difficulty) in a unified representation, without modifying the decoder architecture.

Decoder. The probability that model M_i correctly answers question q_j is estimated as follows:

$$\hat{s}_{ij} = \sigma\left(\phi\left(\mathbf{v}_m^{(i)} \odot \mathbf{v}_q^{(j)}\right)\right), \quad (2)$$

where $\odot$ denotes the Hadamard product, $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^2$ is a linear classifier, and σ applies the sigmoid to the difference between the two output logits.

Training objective. Parameters $\{\mathbf{E}_m, h_{\text{proj}}, \phi, \{\mathbf{e}_\tau\}_{\tau \in \mathcal{T}}\}$ are jointly optimised by minimising binary cross-entropy loss:

$$\mathcal{L} = -\frac{1}{|\mathcal{D}|} \sum_{i,j} [Y_{ij} \log \hat{s}_{ij} + (1 - Y_{ij}) \log (1 - \hat{s}_{ij})]. \quad (3)$$

Relation to matrix factorisation. The system in Equations (1), (2) and (3) extends standard low-rank matrix factorisation (Koren et al., 2009) in two ways: the question factor is not a free parameter but a structured encoding of semantic and task-type content, and the model factor is conditioned on question features via the Hadamard interaction. Unlike scalar item response theory (IRT) models (Vivek et al., 2024), which estimate a single latent ability per model, our embeddings are d -dimensional vectors encoding distinct narrative competence profiles.

4.3 Model Combination Strategies

Once training converges, $\mathbf{E}_m$ and the question encoder are frozen. For a test question q , the vector $\hat{s}_q \in [0, 1]^N$ of predicted correctness scores is computed in a single forward pass. We define three routing strategies (MF-1, MF- k , MF-ctx) and compare them against two baselines (SB, WR) :

MF-1 routes to the highest-scoring model: $M^*(q) = \arg \max_i \hat{s}_{iq}$, the direct narrative-domain adaptation of EmbedLLM (Zhuang et al., 2025). **MF- k** selects the k highest-scoring models and takes a majority vote over their answers; we evaluate $k \in \{3, 5, 10\}$ and select the best k on the validation set. **MF-ctx** is the full model combining MF-1 with complete contextual conditioning ($\mathbf{e}_\tau$ and $\mathbf{f}_j$ in Equation 1). **SB** always routes to the model with the highest average training accuracy, regardless of the question: **WR** routes are randomly weighted by the MF-1 routing distribution, isolating the benefit of question-level discrimination from the effect of preferring better models more frequently.

Model	$\mathbf{e}_\tau$	$\mathbf{f}_j$	Routing	Forecast
MF-1 (base)	×	×	70.4	65.3
MF-1 + task	✓	×	71.1	65.4
MF-1 + features	×	✓	70.9	57.4
MF-ctx(full)	✓	✓	71.1	57.4

Table 3: Ablation of contextual conditioning in the question encoder. $\mathbf{e}_\tau$: task-type embedding. $\mathbf{f}_j$: question features (length, source, difficulty). Routing accuracy and correctness forecasting accuracy (%) on the test set.

Table 3 shows that $\mathbf{e}_\tau$ improves routing by 0.7 points with negligible effect on forecasting, while $\mathbf{f}_j$ introduces a trade-off: routing improves slightly but forecasting accuracy drops from 65.3% to 57.4%, likely because difficulty scores bias the predictor toward the majority class. We retain the full model for routing experiments and report base model results for forecasting.

4.4 Downstream Applications

Beyond routing, frozen embeddings $\mathbf{E}_m$ support two additional tasks with a single linear layer: **correctness forecasting** predicts Y_{ij} for a held-out (model, question) pair from the frozen decoder without fine-tuning; and **benchmark accuracy prediction** trains a linear regression $\mathbf{E}_m \beta \approx \mathbf{a}$ to predict per-dataset accuracy without any additional inference.

5 Experimental Setup

Models. We evaluate 49 open-weight LLMs in a zero-shot manner (50 minus TinyLlama-1.1B, excluded for systematic format failure), spanning 360M–9B parameters across general-purpose, chat-optimised, and one code-specialised model (Qwen2.5-Coder-7B). The full list is in Appendix B.

Inference. Models are evaluated using vLLM (Kwon et al., 2023) (v0.19.1) on NVIDIA L40S 48GB GPUs. Each model receives a narrative-expert system prompt followed by the passage, question, and labelled choices. Generation is constrained to 3 output tokens (1,024 for reasoning models) to enforce single-letter responses; the first letter (A–D) is extracted as the answer.

Embedding training. Questions are encoded with frozen all-mpnet-base-v2 (Reimers and Gurevych, 2019). Embedding dimension d , learning rate, and batch size are tuned across 16 configurations; full details are in Appendix A.

Baselines. Beyond SB and WR (Section 4.3), we compare against **KNN**, which uses each model’s correctness vector on training questions as an implicit representation and classifies via k -nearest neighbours (k tuned on validation).

Metrics. We report **routing accuracy** (the proportion of correctly answered test questions, globally and per dataset), **oracle routing gap** (oracle vs. SB, quantifying headroom for any routing system), **correctness forecasting accuracy** on held-out (model, question) pairs, and **benchmark accuracy prediction** via Kendall’s τ across 100 random model splits.

6 Results

6.1 LLM Performance on NARRATOR

Zero-shot LLM performance is reported in Appendix C across the six datasets. Three observations emerge.

Model size is a poor predictor of narrative competence. Granite-3.1-2B (63.0%) outperforms Yi-1.5-9B (54.5%) overall, and Hermes-3-8B (68.2%) outperforms Llama-3.1-8B (64.2%) despite comparable parameter counts. The Spearman correlation between parameter count and overall accuracy is $\rho = 0.615$ ($p < 0.001$), confirm-

ing that size explains only a fraction of performance variance ($\rho^2 = 0.38$) and that architectural and training differences account for the remainder. Moreover, model family matters more than size: Qwen2.5 scales consistently across size, while Qwen3 slightly underperforms at comparable parameter counts, suggesting that disabling chain-of-thought reasoning removes beneficial implicit narrative reasoning.

Competence profiles are highly heterogeneous.

No model dominates across all datasets. Qwen2.5-Coder-7B leads in NarraDetect (88.3%) despite being code-specialised. However, it ranks 19th overall (60.5%). InternLM2.5-7B reaches 99.1% on ROCStories but ranks 4th overall (67.3%). This heterogeneity motivates sub-task-aware routing: an oracle router achieves 97.1% accuracy compared to 70.5% for the single-best model. For 97.1% of test questions, at least one of the 49 models answers correctly, confirming that the model pool provides complementary coverage across narrative sub-tasks. The gap of 26.3 points between this coverage ceiling and the single-best model (70.5%) quantifies the maximum gain achievable by a perfect per-question router which motivates sub-task-aware model selection.

Annotation quality is consistent with empirical difficulty. MND, with the lowest IAA (Fleiss $\kappa = 0.24$), yields near-random performance across all models (25–37% against a 25% chance baseline). ROCStories, verified to 100% human accuracy, is the easiest dataset (>95% for top models). This directional consistency across the two datasets with reported IAA motivates the geometric analysis in Section 7.3, which provides a richer signal across all six datasets.

6.2 Correctness Forecasting

Table 4 shows that MF base (65.3%) marginally outperforms KNN ($k = 20$, 65.0%), confirming that the learned embeddings capture model-specific correctness patterns beyond simple nearest-neighbour retrieval in question space. However, MF-ctx (57.4%) underperforms both baselines, a consequence of the trade-off identified in Table 3: question features $\mathbf{f}_j$ improve routing but bias the correctness predictor towards the majority class. We therefore report MF base as our primary forecasting result.

Method	Accuracy (%)
KNN ($k = 20$)	65.0
MF base (ours)	65.3
MF-ctx (full)	57.4

Table 4: Correctness forecasting accuracy on held-out model–question pairs.

Method	MND	MPST	NarraDet	ROCSt	StoryDB	Sem26	Overall
SB	31.0	43.0	86.7	95.6	75.2	79.9	70.5
WR	29.8	40.0	77.9	87.2	54.8	67.1	68.5
MF-1	29.6	45.9	86.7	99.2	75.2	81.5	71.1
MF-3	28.0	43.6	87.7	98.3	72.5	84.1	69.2
MF-5	27.6	41.8	87.5	97.8	71.5	82.5	68.2
MF-10	23.6	40.5	86.7	98.1	67.0	81.8	65.0
Oracle	97.0	93.3	99.3	100.0	97.1	100.0	97.1

Table 5: Routing accuracy (%) per dataset and overall.

6.3 Model Routing

Table 5 reports routing accuracy for all strategies on the test set (best hyperparameters: $d = 128$, $\text{lr} = 10^{-4}$, $\text{bs} = 1024$). MF-1 (71.1%) outperforms the single-best router SB (70.5%) and substantially outperforms the weighted random baseline WR (68.5%), confirming that the learned embeddings capture question-level model strengths beyond aggregate accuracy. MF- k strategies underperform MF-1 in this configuration, suggesting that majority voting introduces noise when model competencies are heterogeneous. Per-dataset results range from near-perfect (ROCStories 99.2%) to near-random (MND 29.6%), which is consistent with the annotation quality gradient.

6.4 Benchmark Accuracy Prediction

Table 6 reports Kendall’s τ significance across 100 random model splits. The MF embeddings predict per-dataset accuracy with statistically significant correlation on 91–97% of splits across all six datasets, with mean τ ranging from 0.710 (MPST) to 0.734 (ROCStories). Notably, MND achieves 96% significance despite being the hardest dataset for routing (29.6%). The embeddings capture relative model rankings on MND even when absolute performance is near-random. This suggests that the embedding space encodes latent difficulty structures that are not visible in aggregate accuracy.

7 Probing the Narrative Embedding Space

We conduct a series of probing experiments using the frozen embeddings from the best configuration ($d = 128$, $\text{lr} = 10^{-4}$, $\text{bs} = 1024$).

Dataset	Significance (%)	Mean τ
NarraDetect	93	0.714
MPST	91	0.710
StoryDB	96	0.733
MND	96	0.727
ROCStories	97	0.734
SemEval-2026	93	0.728

Table 6: Benchmark accuracy prediction: proportion of 100 random model splits with statistically significant Kendall’s τ ($p < 0.05$) between predicted and actual per-dataset accuracy. Best configuration: $d = 128$, $lr = 10^{-4}$, $bs = 1024$.

7.1 Structural Validity of Model Embeddings

We verify that models with similar characteristics are closer in ℓ_2 distance than dissimilar ones. We define six binary communities based on parameter count (1B, 3B, 7B+) and model family (Qwen, Llama, Mistral) and compare mean intra-community vs. inter-community ℓ_2 distances.

Table 7 reports the results. Size-based communities (1B, 3B, 7B+) show consistent intra < inter separation (ratios 0.667–0.867), confirming that parameter count organises the embedding space. Family-based separation is more nuanced: Mistral versions cluster tightly (ratio 0.753), while Qwen and Llama models are more dispersed than their inter-community distances (ratios 1.082 and 1.160 respectively), reflecting their wide parameter ranges (0.5B–9B) which introduce more within-family variance than between-family variance.

Community	N	Intra ℓ_2	Inter ℓ_2	Ratio
1B models	16	0.784	0.904	0.867
3B models	11	0.455	0.683	0.666
7B+ models	22	0.548	0.768	0.714
Qwen family	11	0.796	0.736	1.082
Llama family	3	1.000	0.861	1.160
Mistral family	3	0.454	0.604	0.753

Table 7: Mean intra- vs. inter-community ℓ_2 distances. Ratio < 1 indicates models within a community are closer to each other than to outside models. Size-based communities show consistent separation (ratio < 1).

7.2 Task-Type Geometry of Question Embeddings

We probe whether the question embedding space encodes narrative task-type structure by training a linear classifier to predict the dataset source s_j from the frozen question embedding $\mathbf{v}_q^{(j)}$, using 5-fold

cross-validation. The linear probe achieves 99.9% accuracy against a chance level of $1/6 \approx 16.7\%$, confirming that the embedding space separates narrative sub-tasks. Figure 1 shows a t-SNE projection of 5,000 randomly sampled question embeddings, coloured by dataset source.

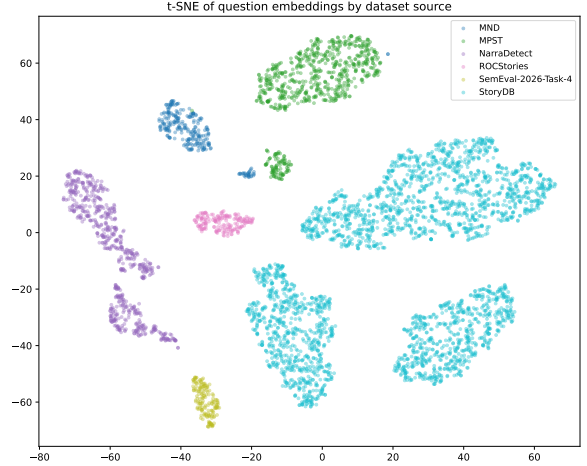


Figure 1: Projection of 5,000 question embeddings coloured by dataset source.

7.3 Annotation Quality as a Geometric Signal

Dataset	Mean ℓ_2	IAA
NarraDetect	0.841	$\alpha = 0.37$
StoryDB	0.539	—
MPST	0.374	—
MND	0.367	$\kappa = 0.24$
SemEval-2026	0.277	Human-annotated
ROCStories	0.197	Human ceiling

Table 8: Mean pairwise ℓ_2 distance between question embeddings within each dataset, sorted by dispersion.

Table 8 shows that NarraDetect has the highest dispersion ($\ell_2 = 0.841$), reflecting its broad domain coverage across 18 genres rather than its moderate IAA alone. ROCStories, with human-verified labels, is the most tightly clustered ($\ell_2 = 0.197$). Interestingly, MND exhibits low dispersion ($\ell_2 = 0.367$) despite its low inter-annotator agreement ($\kappa = 0.24$), suggesting that its questions are semantically similar but unreliably labelled due to annotation noise rather than semantic ambiguity. This finding refines our hypothesis: embedding dispersion reflects *semantic diversity* rather than annotation quality directly, though the two are correlated across most datasets. This reframes MND’s role in NARRATOR: far from being a noisy outlier, it

serves as a controlled calibration case whose geometric isolation reflects annotation noise rather than semantic ambiguity. This distinction would be invisible from agreement scores alone.

7.4 Model Family Structure

Figure 2 shows a t-SNE projection of the 49 model embeddings coloured by model family. Models organise primarily by parameter count rather than family: smaller models cluster toward the bottom-right regardless of architecture. The size-based organisation of the embedding space reflects average behavioural similarity, not per-question predictability; the two are distinct, and it is the latter that routing targets.

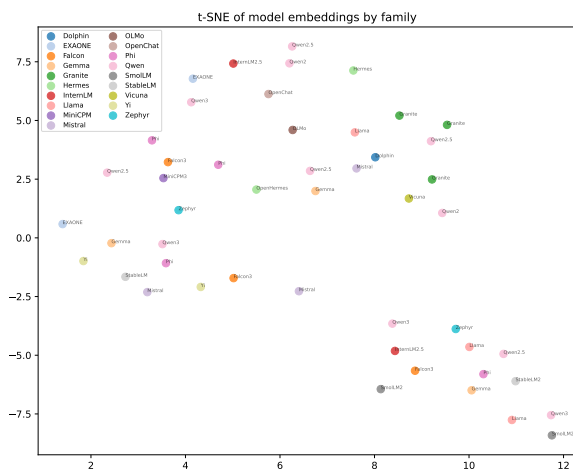


Figure 2: Projection of 49 model embeddings coloured by family.

8 Conclusion

We introduced NARRATOR, a unified benchmark of 96,818 MCQ spanning four sub-tasks and a binary correctness matrix of 50 open-weight LLMs, providing an evaluation resource and a behavioural snapshot of current LLMs on narrative tasks. Our results show that narrative competence profiles are heterogeneous and not predicted by model size, and an oracle routing gap of 26.3 points confirms substantial headroom for sub-task-aware selection. A matrix factorisation framework learns compact narrative competence embeddings from behavioural correctness data alone, supporting routing (71.1% vs. SB 70.5%), forecasting (65.3%), and benchmark accuracy prediction (mean τ 0.71–0.73, significant on 91–97% of splits). Probing shows that task-type structure and annotation quality are geometrically encoded in the embedding space.

NARRATOR, data and code are publicly released.² Future work includes inductive model encoding from small probe sets, without full retraining, the integration of entity and role analysis datasets as a fourth narrative sub-task, cross-dataset transfer analysis, and extension to multilingual and multimodal narratives, a direction we are currently pursuing with a parallel English-Chinese corpus of news narratives.

Limitations

Our model pool is restricted to open-weight models of up to 9B parameters, which represent the most practically deployable range for narrative NLP applications. Extension to larger models (70B+) and to closed-weight APIs is left for future work, as is the question of whether behavioural embedding quality scales with model size.

All models are evaluated under a single system prompt. Prompt sensitivity, which is relevant for smaller and older models with less instruction tuning, is not explored and may affect the correctness matrix and the resulting embeddings. Our training signal is binary (correct/incorrect); richer signals such as calibrated probabilities or human preference judgments could improve embedding expressivity.

The current framework requires retraining when new models are added to the pool, as model embeddings are learned parameters rather than being computed from model weights. NARRATOR is currently limited to written English narratives. Multilingual, visual and spoken narratives are out of scope. Two planned datasets (DIPROMATS, SemEval-2025 Task 10) could not be included due to access restrictions at the time of construction.

Finally, MCQ may underestimate model capabilities on generative narrative tasks such as open-ended comprehension or story generation.

Acknowledgements

This paper was partially funded by the ANR project MILL-EHNAS (Mapping profit Legitimacy Language by Exploring Historical NarrativeS, ANR-25-CE38-6695) as well as by the European Union HORIZON-WIDERA-2023-TALENTS-01-01 grant 101186647 — AI4DH. Views and opinions expressed are those of the author(s) only and do not necessarily reflect those of the European

²<https://github.com/ahHamdi/narrator>.

Union. Neither the European Union nor the granting authority can be held responsible for them.

References

- Lingjiao Chen, Matei Zaharia, and James Zou. 2023. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*.
- Yew Ken Chia, Pengfei Hong, Lidong Bing, and Soujanya Poria. 2024. Instructeval: Towards holistic evaluation of instruction-tuned large language models. In *Proceedings of the First edition of the Workshop on the Scaling Behavior of Large Language Models (SCALE-LLM 2024)*, pages 35–64.
- Jesús M Fraile-Hernández, Anselmo Peñas, and Pablo Moral. 2024. Automatic identification of narratives: Evaluation framework, annotation methodology, and dataset creation. *IEEE Access*, 13:11734–11753.
- Sil Hamilton, Matthew Wilkens, and Andrew Piper. 2026. Narrabench: A comprehensive framework for narrative benchmarking. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3786–3801.
- Hans Ole Hatzel, Ekaterina Artemova, Haimo Stierner, Evelyn Gius, and Chris Biemann. 2026. SemEval-2026 Task 4: Narrative similarity and narrative representation learning. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, San Diego, CA, USA. Association for Computational Linguistics.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.
- David Herman. 2017. *Storytelling and the Sciences of Mind*. MIT press.
- Sudipta Kar, Suraj Maharjan, A Pastor López-Monroy, and Thamar Solorio. 2018. Mpst: A corpus of movie plot synopses with tags. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Sebastien Kawada and Dylan Holyoak. 2026. Cascademind at semeval-2026 task 4: A hybrid neuro-symbolic cascade for narrative similarity. In *Proceedings of the 20th International Workshop on Semantic Evaluation (2026)*, pages 1575–1581.
- Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.
- Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yan Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.
- Chang Liu, Armin Shmilovici, and Mark Last. 2022. Mnd: A new dataset and benchmark of movie scenes classified by their narrative function. In *European Conference on Computer Vision*, pages 610–626. Springer.
- Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849.
- Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. 2024. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*.
- Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, Mohammed Kadous, and Ion Stoica. 2025. Routellm: Learning to route llms from preference data. In *International Conference on Learning Representations*, volume 2025, pages 34433–34448.
- Andrew Piper and Sunyam Bagga. 2025. Narradetect: An annotated dataset for the task of narrative detection. In *Proceedings of the The 7th Workshop on Narrative Understanding*, pages 1–7.
- Jakub Piskorski, Tarek Mahmoud, Nikolaos Nikolaidis, Ricardo Campos, Alipio Mario Jorge, Dimitar Dimitrov, Purificação Silvano, Roman Yangarber, Shivam Sharma, Tanmoy Chakraborty, and 1 others. 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)*, pages 2610–2643.
- Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. 2024. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical*

methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pages 3982–3992.

Marija Šakota, Maxime Peyrard, and Robert West. 2024. Fly-swat or cannon? cost-effective language model choice via meta-modeling. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 606–615.

Tal Shnitzer, Anthony Ou, Mírian Silva, Kate Soule, Yuekai Sun, Justin Solomon, Neil Thompson, and Mikhail Yurochkin. 2023. Large language model routing with benchmark datasets. *arXiv preprint arXiv:2309.15789*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, and 1 others. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Alexey Tikhonov, Igor Samenko, and Ivan P Yamshchikov. 2021. Storydb: Broad multi-language narrative dataset. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 32–39.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. 2024. Anchor points: Benchmarking models with much fewer examples. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1576–1601.

Zihao Yue, Yepeng Zhang, Ziheng Wang, and Qin Jin. 2025. Movie101v2: Improved movie narration benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17081–17095.

Yuxiang Zeng, Yongxin Tong, and Lei Chen. 2023. Litehst: A tree embedding based method for similarity search. *Proceedings of the ACM on Management of Data*, 1(1):1–26.

Richard Zhuang, Tianhao Wu, Zhaojin Wen, Andrew Li, Jiantao Jiao, and Kannan Ramchandran. 2025. Embedllm: Learning compact representations of large language models. In *International Conference on Learning Representations*, volume 2025, pages 76913–76926.

A Hyperparameter Details

We explored 16 combinations of hyperparameters as indicated in Table 9 and selected best configuration on validation using MF-1. Full results are presented Table 10.

Hyperparameter	Values / Setting
Embedding dim d	{32, 64, 128, 256}
Learning rate	{ 10^{-3} , 10^{-4} }
Batch size	{1024, 2048}
Task emb. dim d_τ	16
Question feature dim d_f	10
Optimiser	Adam
Weight decay	10^{-4}
Early stopping patience	5 epochs
Max epochs	50
ST model	all-mpnet-base-v2

Table 9: Hyperparameter grid and fixed settings.

d	lr	bs	MF-1	τ mean
32	10^{-3}	1024	69.7	0.519
32	10^{-3}	2048	70.6	0.507
32	10^{-4}	1024	70.4	0.510
32	10^{-4}	2048	70.4	0.523
64	10^{-3}	1024	70.7	0.543
64	10^{-3}	2048	70.8	0.602
64	10^{-4}	1024	70.6	0.648
64	10^{-4}	2048	70.4	0.655
128	10^{-3}	1024	70.1	0.578
128	10^{-3}	2048	68.5	0.658
128	10^{-4}	1024	71.1	0.724
128	10^{-4}	2048	70.5	0.750
256	10^{-3}	1024	69.6	0.533
256	10^{-3}	2048	70.2	0.581
256	10^{-4}	1024	70.7	0.778
256	10^{-4}	2048	70.5	0.784

Table 10: Grid search results across 16 configurations. Bold: best MF-1 routing accuracy and best mean Kendall’s τ . The selected configuration ($d = 128$, lr= 10^{-4} , bs= 1024) maximises routing accuracy.

B Full Model List

The detailed list of LLMs, number of params, family, and type are presented in Table 11.

Model	Params	Family	Type
Qwen2.5-7B-Instruct	7B	Qwen	General
Qwen2-7B-Instruct	7B	Qwen	General
Qwen2.5-3B-Instruct	3B	Qwen	General
Qwen2.5-1.5B-Instruct	1.5B	Qwen	General
Qwen2.5-0.5B-Instruct	0.5B	Qwen	General
Qwen2-1.5B-Instruct	1.5B	Qwen	General
Qwen2.5-Coder-7B	7B	Qwen	Code
Qwen3-8B	8B	Qwen	General
Qwen3-4B	4B	Qwen	General
Qwen3-1.7B	1.7B	Qwen	General
Qwen3-0.6B	0.6B	Qwen	General
Llama-3.1-8B-Instruct	8B	Llama	General
Llama-3.2-3B-Instruct	3B	Llama	General
Llama-3.2-1B-Instruct	1B	Llama	General
Mistral-7B-v0.1	7B	Mistral	General
Mistral-7B-v0.2	7B	Mistral	General
Mistral-7B-v0.3	7B	Mistral	General
Granite-3.1-2B	2B	Granite	General
Granite-3.2-2B	2B	Granite	General
Granite-3.2-8B	8B	Granite	General
Phi-3-mini-4k	3.8B	Phi	General
Phi-3.5-mini	3.8B	Phi	General
Phi-4-mini	3.8B	Phi	General
Phi-1.5-1.3B	1.3B	Phi	General
Gemma-2-2B-IT	2.7B	Gemma	General
Gemma-3-4B-IT	4B	Gemma	General
Gemma-3-1B-IT	1B	Gemma	General
InternLM2.5-7B	7B	InternLM	General
InternLM2.5-1.8B	1.8B	InternLM	General
EXAONE-3.5-7.8B	7.8B	EXAONE	General
EXAONE-3.5-2.4B	2.4B	EXAONE	General
Hermes-3-8B	8B	Hermes	Chat
OpenHermes-2.5-7B	7B	Hermes	Chat
OpenChat-3.5-7B	7B	OpenChat	Chat
Dolphin-2.9-7B	7B	Dolphin	Chat
Vicuna-7B-v1.5	7B	Vicuna	Chat
Zephyr-7B-beta	7B	Zephyr	Chat
Zephyr-7B-Gemma	7B	Zephyr	Chat
Falcon3-7B-Instruct	7B	Falcon	General
Falcon3-3B-Instruct	3B	Falcon	General
Falcon3-1B-Instruct	1B	Falcon	General
Yi-1.5-9B-Chat	9B	Yi	Chat
Yi-1.5-6B-Chat	6B	Yi	Chat
OLMo-2-7B	7B	OLMo	General
MiniCPM3-4B	4B	MiniCPM	General
SmoLM2-1.7B	1.7B	SmoLLM	General
SmoLM2-360M	0.36B	SmoLLM	General
StableLM-Zephyr-3B	3B	StableLM	Chat
StableLM2-Zephyr-1.6B	1.6B	StableLM	Chat

Table 11: The 49 LLMs evaluated in NARRATOR zero-shot setting, grouped by model family. Parameter counts are approximate. TinyLlama-1.1B is excluded due to systematic format-following failure (generates verbose multi-token outputs instead of single letters).

C Zero Shot Accuracy

Accuracy performances for all models using the six datasets are presented in Table 12.

D NARRATOR Samples

In this section, we present examples of the multiple-choice transformation of the NARRATOR benchmark, illustrated for each of the six source datasets: NarraDetect (Table 13), MPST (Table 14), StoryDB (Table 15), MND (Table 16), ROCStories (Table 17), and SemEval-2026 (Table 18).

E MCQ Transformation Examples

Table 19 illustrates the transformation pipeline for two datasets. Each item shows the original format alongside the resulting NARRATOR MCQ item.

F Per-Model Format Failure Rates

Table 20 reports the percentage of -1 labels (failed format responses) per model. TinyLlama-1.1B is excluded entirely due to systematic failure. For the remaining 49 models, failure rates are sparse (almost $< 1\%$) and show no systematic pattern correlated with question difficulty or dataset source.

Model	-1 rate (%)
InternLM2.5-1.8B	0.003%
InternLM2.5-7B	0.001%
Vicuna-7B-v1.5	0.002%
Phi-1.5-1.3B	1.217%
Mistral-7B-v0.1	0.095%
Mistral-7B-v0.2	0.476%
Mistral-7B-v0.3	0.005%
MiniCPM3-4B	0.001%
OpenChat-3.5-7B	0.031%
StableLM2-Zephyr-1.6B	4.869%
OpenHermes-2.5-7B	0.212%

Table 20: Per-model format failure rates (-1 labels).

Model	Params	MND	MPST	NarDet	ROCSt	Sem26	StoryDB	Overall
Qwen2.5-7B-Instruct	7B	35.8	43.4	87.5	95.9	84.4	74.9	70.9
Qwen2-7B-Instruct	7B	29.4	42.6	86.1	95.6	85.2	72.0	68.5
Hermes-3-8B	8B	32.9	44.6	81.0	95.7	77.4	72.3	68.2
InternLM2.5-7B	7B	31.5	46.3	86.1	99.1	85.1	68.5	67.3
EXAONE-3.5-7.8B	7.8B	31.6	44.0	87.0	98.3	82.2	65.4	65.1
OpenChat-3.5-7B	7B	34.1	45.9	83.2	98.9	86.0	65.4	65.1
Llama-3.1-8B-Instruct	8B	30.7	43.6	83.8	96.6	59.3	66.0	64.2
Qwen3-8B	8B	29.7	41.7	87.5	96.4	79.7	64.4	64.0
OLMo-2-7B	7B	26.7	44.7	85.8	90.8	80.3	64.6	63.9
Granite-3.1-2B	2B	28.4	39.8	81.9	91.6	64.8	65.7	63.0
Granite-3.2-2B	2B	29.0	40.3	78.5	91.0	60.5	65.9	62.6
Qwen2.5-3B-Instruct	3B	33.3	41.6	78.0	84.2	69.6	65.1	62.5
Phi-3-mini-4k	3.8B	30.7	44.0	87.1	97.0	81.3	60.3	62.0
Dolphin-2.9-7B	7B	31.1	39.0	77.4	96.3	74.4	64.0	61.9
Falcon3-7B-Instruct	7B	27.9	47.1	85.7	96.5	83.2	59.1	61.5
Mistral-7B-v0.1	7B	31.9	44.4	79.5	93.8	62.4	62.1	61.4
Phi-3.5-mini	3.8B	30.6	48.9	84.5	96.5	69.5	58.8	61.1
Qwen2.5-1.5B-Instruct	1.5B	33.6	48.4	75.7	91.5	76.0	60.4	60.9
Qwen2.5-Coder-7B	7B	36.9	44.6	88.3	95.8	80.6	56.8	60.5
MiniCPM3-4B	4B	34.4	41.7	85.1	95.8	83.5	57.8	60.1
Gemma-2-2B-IT	2.7B	30.9	45.3	77.7	89.0	67.2	58.3	59.0
OpenHermes-2.5-7B	7B	33.8	44.5	79.8	96.1	83.9	55.7	58.6
Vicuna-7B-v1.5	7B	32.3	39.3	75.5	74.1	68.3	60.1	58.4
Granite-3.2-8B	8B	33.4	39.1	74.0	96.9	69.0	58.3	58.1
Zephyr-7B-beta	7B	32.0	42.6	85.4	93.1	68.9	54.5	57.7
Gemma-3-4B-IT	4B	32.3	40.0	87.5	94.4	82.1	52.6	57.0
Qwen3-4B	4B	31.3	35.0	85.6	92.5	78.8	53.6	56.3
Qwen2-1.5B-Instruct	1.5B	30.7	36.7	62.3	92.6	72.7	58.5	56.0
InternLM2.5-1.8B	1.8B	35.4	46.3	64.8	89.9	66.7	54.9	55.6
Qwen3-1.7B	1.7B	31.2	40.1	80.5	81.6	57.1	54.0	55.5
EXAONE-3.5-2.4B	2.4B	26.6	36.2	86.7	96.6	79.2	51.5	55.3
Phi-4-mini	3.8B	28.9	43.1	82.6	95.7	80.3	49.9	54.9
Zephyr-7B-Gemma	7B	33.7	40.0	80.7	94.7	60.3	51.5	54.8
Yi-1.5-9B-Chat	9B	34.0	39.2	87.1	96.2	81.8	48.4	54.5
Falcon3-3B-Instruct	3B	33.7	38.0	75.2	91.6	75.0	51.4	54.0
SmolLM2-1.7B	1.7B	27.9	28.6	61.9	82.5	50.1	58.5	53.5
Mistral-7B-v0.2	7B	32.4	48.3	85.2	89.7	55.8	46.1	53.1
StableLM-Zephyr-3B	3B	29.1	40.8	83.0	90.8	71.1	47.5	52.8
Mistral-7B-v0.3	7B	33.4	40.1	74.8	94.3	54.5	49.6	52.7
Yi-1.5-6B-Chat	6B	25.3	36.2	74.9	93.7	79.5	49.8	52.5
Llama-3.2-3B-Instruct	3B	25.7	33.7	71.1	89.4	53.8	46.1	48.5
Gemma-3-1B-IT	1B	25.0	42.5	64.9	64.5	50.3	44.8	47.1
Falcon3-1B-Instruct	1B	24.8	35.8	62.4	77.4	57.9	42.2	44.9
Phi-1.5-1.3B	1.3B	30.4	36.4	61.8	83.7	30.4	42.3	44.7
Qwen2.5-0.5B-Instruct	0.5B	30.5	39.2	61.8	55.6	51.2	42.0	44.5
Llama-3.2-1B-Instruct	1B	23.4	40.5	61.9	70.4	49.7	40.1	43.7
Qwen3-0.6B	0.6B	24.5	31.1	82.5	64.5	50.1	37.6	43.6
StableLM2-Zephyr-1.6B	1.6B	28.4	29.9	68.2	55.0	56.9	39.2	42.4
SmolLM2-360M	0.36B	25.1	26.2	78.0	51.6	47.0	26.6	35.2

Table 12: Zero-shot accuracy (%) of 49 LLMs on NARRATOR, sorted by overall accuracy. **Bold**: best per column. MND performance is near-random for all models, consistent with its low inter-annotator agreement ($\kappa = 0.24$).

Task: Binary — is the text a narrative or not?

Text: "Confidentiality. Concurrently with this Agreement, the parties shall enter into a confidentiality agreement (the "Confidentiality Agreement"), a copy of which has been attached hereto as Exhibit A, which shall govern the confidentiality of any Confidential Information..."

Question: Is this text a narrative?

- A) Yes, this text contains a narrative with story elements
 - B) No, this is expository or non-narrative text
-

Table 13: Random sample of the NarraDetect in the MCQ format. Correct answer is B for this example.

Task: Identify the primary genre tag of a movie plot synopsis (4 choices, 3 distractors)

Text: "In 1969, ten-year-old Frank Sears is confronted by his hysterical mother Eleanore, who carves the message 'S3 V12' into the palms of his hands with a pencil and warns him to never forget her warning before murdering her husband and committing suicide..."

Question: What is the primary genre or theme of this movie plot?

- A) murder
 - B) psychedelic
 - C) romantic
 - D) revenge
-

Table 14: Random sample of the MPST in the MCQ format. Correct answer is A for this example.

Task: Identify the genre of a Wikipedia story plot (4 choices, 3 distractors).

Text: "<p>The story follows survivors dealing with a pandemic that causes its victims to carry out their most evil thoughts. Carriers of the virus are generally known as the 'Crossed' due to a large, cross-like rash that appears on their faces..."

Question: What is the genre of this story?

- A) film
 - B) comedy
 - C) horror
 - D) television
-

Table 15: Random sample of the StoryDB in the MCQ format. Correct answer is C for this example.

Task: Identify the Save the Cat story beat for a movie scene (4 choices). Scene dialogue comes from OpenSubtitles '.srt' files aligned to MND scene timestamps (42/45 movies covered; 3 fall back to scene feature metadata).

Text: "[Harry Potter And The Sorcerer's Stone, scene 174] Take care of Ron. Then go to the owlery. Send a message to Dumbledore. Ron's right. I have to go on. You'll be okay, Harry. You're a great wizard. You really are. Not as good as you. Me? Books and cleverness. There are more important things. Friendship and bravery. And, Harry, just be careful."

Context: Harry Potter And The Sorcerer's Stone

Question: What is the narrative function of this movie scene?

- A) Finale
 - B) Final Image
 - C) Break Into Three
 - D) Dark Night of the Soul
-

Table 16: Random sample of the MND in the MCQ format. Correct answer is A for this example.

Task: Choose the correct ending to a 4-sentence story (2 choices).

Text: "Katie decided it would be fun to dye some Easter eggs. She got her siblings together and they started dying a dozen. They chose all different colors. When they were done they looked beautiful."

Question: Which ending correctly completes this story?

- A) They were proud of their eggs.
- B) They were sad about the eggs.

Table 17: Random sample of the ROCStories in the MCQ format. Correct answer is A for this example.

Task: Which of two candidate stories is more narratively similar to an anchor story (2 choices)?

Anchor: "Grammy Award-winning music act Milli Vanilli become the first act to have their prize stripped after being exposed as a fake enterprise with lip-syncing singers. The movie follows their meteoric rise to one of the most successful pop duos..."

Question: Which story (A or B) is more narratively similar to the anchor story?

- A) Story A: "The landowner Philipp Klapproth finances his nephew's medical studies. The truth is, the nephew has completely different plans he can neither see blood, nor has he ever studied medicine..."
- B) Story B: "In 1927, unemployed actor Harry Frommermann creates a German singing group. They meet international fame but eventually run into trouble when the Nazis come to power..."

Table 18: Random sample of the SemEval-2026 Task 4 in the MCQ format. Correct answer is A for this example.

Original format	NARRATOR MCQ
<i>MPST (illustrative)</i> plot_synopsis: "Jules and Vincent, two hitmen..." tags: cult, violence, nonlinear, dark	Text: "Jules and Vincent, two hitmen..." question: What is the primary genre or theme? choices: [A. cult, B. romance, C. dark, D. biography] answer: A (cult)
<i>MND (illustrative)</i> movie: Pulp Fiction, scene: 12 beat_label: Set-Up features: dur=45.3s, shots=12	Text: "{Extracted from OpenSubtitles str files using scene-timestamp information}..." question: What is the narrative function of this scene? choices: [A. Midpoint, B. Set-Up, C. Finale, D. Bad Guys Close In] answer: B (Set-Up)

Table 19: Illustrative MCQ transformation examples for MPST and MND. Original labels are preserved in metadata.