What Really Controls Temporal Reasoning in Large Language Models: Tokenisation or Representation of Time?

Anonymous ACL submission

Abstract

We present MULTITEMPBENCH, a multilingual temporal reasoning benchmark spanning three tasks, date arithmetic, time zone conversion, and temporal relation extraction across five languages (English, German, Chinese, Arabic, and Hausa) and multiple calendar conventions (Gregorian, Hijri, and Chinese Lunar). MULTITEMPBENCH contains 15,000 examples built by translating 750 curated English questions and expanding each into controlled date-format variants. We evaluate 20 LLMs and introduce the *multilingual Date Fragmentation Ratio* (mDFR), calibrated with human severity ratings, together with geometric-probing analyses of internal temporal representations. We find that multilingual temporal failures are not explained by resource level alone: the dominant bottleneck shifts by regime. In low-resource languages and rarer calendar formats, fragmentation disrupts access to Year/Month/Day components and strongly predicts accuracy collapse, whereas in high-resource languages, models are often robust to similar fragmentation and performance is better predicted by the linear decodability of temporal values in hidden representations.

1 Introduction

Time is a universal substrate for human reasoning, but temporal expressions are deeply *language*- and *culture*-specific. Real-world systems—calendar assistants, travel planners, clinical and legal timeline reconstruction, historical question answering, and forecasting—must interpret and manipulate dates, times, and temporal relations expressed in heterogeneous surface forms (e.g., `2024-05-01` vs. "1 May 2024") and under distinct calendrical conventions (e.g., Gregorian vs. Hijri vs. Lunar). These requirements are inherently multilingual: users routinely mix scripts, localized month lexemes, and calendar markers, and many high-stakes workflows depend on correct temporal normalisation and arithmetic across languages and regions. Recent temporal benchmarks have advanced our understanding of LLMs' abilities in date arithmetic, temporal ordering, and time-sensitive QA, but they overwhelmingly focus on English and Gregorian representations (Wang and Zhao, 2024; Fatemi et al., 2024; Zhu et al., 2024; Chu et al., 2023; Islakoglu and Kalo, 2025; Wei et al., 2025; Liu et al., 2025; Sasaki et al., 2025; Pęzik et al., 2025). In parallel, emerging work on *cross-calendar* reasoning highlights that LLMs remain inadequate for inter-calendar conversion and that non-Gregorian temporal structure is still underexplored despite its global relevance (Han et al., 2025; Saxena et al., 2025; Miao et al., 2025a; Wang and Dong, 2026; Holtermann et al., 2025).

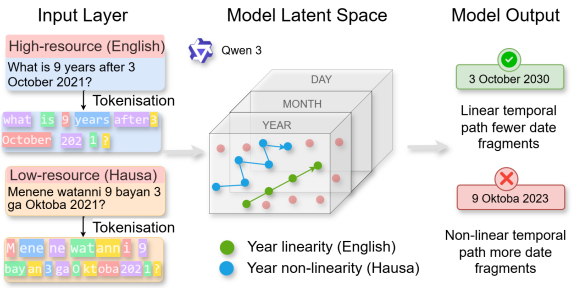


Figure 1: **Mechanistic understanding of multilingual temporal reasoning in MULTITEMPBENCH.**

An orthogonal challenge concerns *how dates are presented to the model*. Temporal strings are structured symbolic objects, yet subword tokenisers (e.g., BPE and byte-level tokenisation) can fragment them into opaque substrings, potentially erasing semantic boundaries such as year/month/day separators and calendar markers (Spathis and Kawsar, 2023; Bhatia et al., 2025a). More broadly, multilingual tokenisation is known to induce systematic inequities: low-resource languages often incur heavier fragmentation (a "token tax"), which increases effective sequence length and can degrade downstream performance (Ahia et al., 2023;

Petrov et al., 2023; Lundin et al., 2025a; Kanjirangat et al., 2025). For numeracy specifically, the choice between digit-level tokenisation and larger numeric chunks yields distinct arithmetic failure modes: inconsistent segmentation forces the model to infer place value and grouping from unstable boundaries, complicating the learning of multi-digit operations and carry-like mechanisms (Singh and Strouse, 2024; Kreitner et al., 2025). These observations motivate a natural hypothesis: multilingual temporal failures may be primarily a *tokenisation problem*. However, two gaps prevent a mechanistic account. First, existing tokenisation analyses typically target general text or broad downstream tasks, rather than *controlled temporal expressions* that combine digits, delimiters, month lexemes, and calendar markers across scripts and calendar systems. Even temporally focused studies largely remain monolingual or calendar-specific (Bhatia et al., 2025a; Han et al., 2025; Miao et al., 2025a). Second, even when behavioural gaps are documented (e.g., accuracy differences and systematic error patterns across languages and formats), we lack clarity on *where* the failure arises in the processing pipeline: does temporal information degrade at the input layer, fail to form an abstract representation suitable for computation, or break during reasoning and decoding? Prior mechanistic work suggests that LLMs can encode ordered scalar attributes in ways that are approximately linearly decodable from hidden states, and can exhibit stable latent directions corresponding to monotonic temporal progression (Gurnee and Tegmark, 2024; El-Shangiti et al., 2025), but these findings have not been connected to multilingual, multi-calendar temporal reasoning in a controlled setting. As a result, it remains unclear whether multilingual temporal competence requires (i) better surface segmentation, (ii) a shared internal "calendar geometry" that supports computation, or (iii) both.

Our contributions are summarized as follows:

(i) **A controlled multilingual, multi-calendar benchmark.** We release MULTITEMPBENCH, comprising 15,000 examples across 5 languages, 3 temporal tasks, and multiple date-format complexity levels, including Gregorian, Hijri, and Lunar calendar systems. We evaluate a broad set of LLMs (open-weight and proprietary) in a zero-shot setting, quantifying how language resource level, format complexity, and calendar system affect temporal reasoning.

(ii) **A multilingual fragmentation metric.** We propose **mDFR**, a multilingual extension of Date Fragmentation Ratio (Bhatia et al., 2025a) that penalises semantically destructive segmentations (e.g., digit splitting and boundary loss) with weights learned from human severity ratings.

(iii) **A mechanistic account of multilingual temporal performance gaps.** We show that bottlenecks shift by language: *date fragmentation* is most predictive of failure in low-resource regimes, whereas *temporal linearity* (probe R^2) is the strongest predictor of temporal task performance in high-resource languages once calendar components (e.g., Year/Month/Day) are frequently accessible in training data. This supports a two-stage view in which tokenisation governs surface access to calendar components, while temporal linearity governs internal temporal representations. We test this with a crossed mixed-effects regression over all model predictions.

2 Related Works

Tokenisation bias in multilingual models. Tokenisation remains a critical source of disparity in multilingual LLMs. Recent studies confirm that low-resource languages and dialects suffer a token tax, inflated sequence lengths that degrade performance and increase compute costs (Lundin et al., 2025a; Kanjirangat et al., 2025). Such fragmentation may critically impact numeric and temporal reasoning. Bhatia et al. (2025a); Karthika et al. (2025) identify date fragmentation as a hidden bottleneck, showing that Byte-Pair Encoding (BPE) often splits dates into opaque substrings that hinder temporal arithmetic. Similarly, Singh and Strouse (2024) demonstrate that standard tokenisation degrades arithmetic performance compared to single-token number embeddings. Our work extends this line by isolating how multilingual date-tokenisation quality, measured via mDFR, affects temporal reasoning.

Mechanisms of time: memorisation vs. reasoning. Evaluating temporal understanding requires distinguishing between pattern matching and robust reasoning. Indeed, recent studies reveal that while LLMs maintain stable performance on memorisation-based temporal tasks, their accuracy sharply declines on reasoning-intensive tasks, especially when navigating temporal shifts or inte-

grating new knowledge (Mazzia et al., 2026; Li and Goyal, 2025; Li et al., 2025). Benchmarks such as ChronoSense (Islakoglu and Kalo, 2025), SPAN (Miao et al., 2025b), DateLogicQA (Bhatia et al., 2025b) and TimeBench (Chu et al., 2023) reveal that LLMs struggle with symbolic constraints and temporal commonsense. Mechanistically, Gurnee and Tegmark (2024) find that models possess linear subspaces representing space and time, suggesting relevant information is encoded but not always exploited. However, Mamidanna et al. (2025) observe that computation is often aggregated only at the final token, creating a fragile information bottleneck. We situate MULTITEMPBENCH at the intersection of these fields. Unlike broad benchmarks, we use controlled date expressions to disentangle tokenisation from reasoning.

3 Our MULTITEMPBENCH

Lang. (Size)	Type	Pattern	Example
English (300GB)	ISO	YYYY-MM-DD	2023-07-03
	Numeric	DD/MM/YYYY	03/07/2023
	Textual	DD Month YYYY	03 July 2023
	Phrasal	Day of Month YYYY	3rd of July 2023
German (66GB)	ISO	YYYY-MM-DD	2023-07-03
	Numeric	DD.MM.YYYY	03.07.2023
	Textual	DD. Month YYYY	03. Juli 2023
	Phrasal	DD. Mon... YYYY	03. Juli des Jahres 2023
Chinese (47GB)	ISO	YYYY-MM-DD	2023-07-03
	Numeric	DD/MM/YYYY	03/07/2023
	Textual	Y 年M 月D 日	2023年07月03日
	Lunar	*Traditional*	二零二三年六月初九
Arabic (28GB)	ISO	YYYY-MM-DD	2023-07-03
	Numeric	DD/MM/YYYY	03/07/2023
	Textual	DD Month YYYY	٣ يوليو ٣٢٠٢
	Hijri	Hijri DD Mon YYYY	٣ ربيع الأول ٥٠٤هـ
Hausa (0.3GB)	ISO	YYYY-MM-DD	2023-07-03
	Numeric	DD/MM/YYYY	03/07/2023
	Textual	DD ga Month YYYY	03 ga Yuli 2023
	Hijri	DD Mon YYYY AH	03 Ramadan 1445 AH

Table 1: **Date formats and calendar systems in MULTITEMPBENCH.** Values in parentheses after each language denote approximate CommonCrawl-100 language-resource size, not benchmark size. Each language is expanded into exactly four date-format variants. The "Type" column indicates the format category or specific calendar system (e.g., Lunar, Hijri); all other entries use the Gregorian calendar.

Dataset construction. We introduce MULTITEMPBENCH, a multilingual temporal reasoning benchmark derived from three existing datasets: TRAM (Wang and Zhao, 2024), ToT (Fatemi et al., 2024), and FreshBench (Zhu et al., 2024). TRAM contains 526,668 multiple-choice questions across 10 temporal reasoning tasks covering the period from 1000 to 2024. ToT consists of 46,480 questions focusing on temporal semantics and arithmetic from 52 AD to 2087. FreshBench provides 4,643 forecasting questions from 1900 to 2025. To construct the English foundation of MULTITEMPBENCH, we curated a balanced subset of 750 questions: 250 from TRAM, 250 from ToT, and 250 from FreshBench, covering three temporal reasoning tasks: (i) **Date Arithmetic**, which evaluates the ability of LLMs to perform addition and subtraction on dates; (ii) **Time Zone Conversion**, which tests the understanding of LLMs to calculate time differences between regions and (iii) **Temporal Relation**, which infers the relationship (e.g., before, after, simultaneous) between a specific event and a reference date. Data samples in these tasks are provided in Table 5 (appendix). We selected questions where date components (year, month, and day) are fully specified, then we preprocessed them to remove synthetic entities (e.g., "E15") and internal prompting instructions, ensuring all questions are grammatically correct and natural.

Multilingual extension. We extended these 750 English questions into four additional languages: German, Chinese, Hausa, and Arabic, using the Google Translate (Comanici et al., 2025). We manually verified the machine-generated translations. For each target language, two native speakers validated the translations and edited them when necessary to ensure that both the linguistic content and date formats were error-free. The set of languages we selected is based on our linguistic expertise, as well as diversified data availability on the CommonCrawl-100 corpus (Suárez et al., 2019; Penedo et al., 2025), ranging from high-resource languages like English (300 GB) to low-resource ones like Hausa (0.3 GB). As detailed in Table 1, these languages also cover three calendar systems: Gregorian, Hijri, and Chinese Lunar for temporal reasoning.

Data format extension. To assess robustness of temporal reasoning across date formats, we utilized a template-based approach to expand each question into four variants per language with increasing levels of complexity. As shown in Table 1, these formats range from *standard ISO Numeric* (e.g., YYYY-MM-DD) to *Localised Numeric* formats using local separators, and finally to *Calendar-specific* phrases (e.g., "03 Ramadan 1445 AH" for Hausa). For calendar-specific variants (e.g., Hijri and Chinese Lunar), we converted to target-language calendars by using existing calendar con-

version tools (Alshehri, 2024); the results were verified by native speakers. This expansion results in 3,000 questions per language, totalling 15,000 questions. For a detailed description of the conversion tools, library specifications, and language-specific formatting rules, we refer to Appendix A.1.

4 Our Approach

Our aim is to identify the underlying factors that control temporal reasoning, then evaluate how these factors vary across languages. To do so, we present a metric, which we call the multilingual Date Fragmentation Ratio (mDFR), to measure the tokenisation quality of dates, then introduce temporal geometry to capture the geometric structures of internal temporal representations.

4.1 Multilingual Date Fragmentation Ratio

We extend the Date Fragmentation Ratio (DFR) from Bhatia et al. (2025a) to a multilingual setup, which we call mDFR denoted as $F \in [0, 1]$:

$$F = \alpha_1 \mathbb{1}_{\text{split}} + \alpha_2 \mathbb{1}_{\text{delimiter}} + \alpha_3 \Delta N + \alpha_4 \theta \quad (1)$$

Here, $\mathbb{1}_{\text{split}}$ and $\mathbb{1}_{\text{delimiter}}$ are binary indicators for split semantic roots (e.g., splitting "2024") and lost separators, respectively. ΔN is token-count inflation relative to a semantic baseline, and θ is the cosine distance between the model token-count vector and the ideal semantic-unit vector. We calibrate α by fitting a linear model from these four features to average human fragmentation-severity ratings; Appendix A.2 reports high annotator agreement and strong correlation with human ratings.

4.2 Temporal Geometry

Embedding extraction. For each language $\ell \in \{\text{EN}, \text{DE}, \text{ZH}, \text{AR}, \text{HA}\}$ and year $y \in [1990, 2024]$ in 3 different date formats (ISO, Slash and Long) we aim to extract a robust representation of the year that is invariant to specific months or days. To do so, we sample $K = 5$ distinct full dates within year y (e.g., "1995-03-12", "1995-11-05") and embed them into declarative templates (e.g., "The date is <*date*>" or التاريخ هو <*date*>). We propagate these sequences through the model and extract the hidden state $\mathbf{h}^{(\ell)}_{y,k,i} \in \mathbb{R}^d$ corresponding to the final token at layer i for the k-th date sample of year y. We define the average embedding as:

$$\bar{\mathbf{h}}^{(\ell)}_{y,i} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{h}^{(\ell)}_{y,k,i}. \quad (2)$$

Geometric notations. We use several geometric concepts to describe the geometry of time in the embedding space.

- **Line segment.** $\mathbf{s}^{(\ell)}_{y,i}$ is a line segment connecting two vectors $\bar{\mathbf{h}}^{(\ell)}_{y+1,i}$ and $\bar{\mathbf{h}}^{(\ell)}_{y,i}$, indicating the vector difference between two years $y{+}1$ and y in the embedding space:

$$\mathbf{s}^{(\ell)}_{y,i} = \bar{\mathbf{h}}^{(\ell)}_{y+1,i} - \bar{\mathbf{h}}^{(\ell)}_{y,i}. \quad (3)$$

- **A path of line segments.** A sequence of line segments is denoted as

$$\mathcal{P}^{(\ell)}_i = \left(\mathbf{s}^{(\ell)}_{y_1,i}, \mathbf{s}^{(\ell)}_{y_2,i}, \ldots, \mathbf{s}^{(\ell)}_{y_T,i}\right)$$

  where each line segment connects to the next, forming a path of years from 1 to T in the embedding space.

- **The path direction.** We denote the overall path direction as the average of line segments:

$$\mathbf{\Delta}^{(\ell)}_i = \frac{1}{|Y| - 1} \sum_{y} \mathbf{s}^{(\ell)}_{y,i}. \quad (4)$$

  If most line segments point in the same direction, then $\mathbf{\Delta}^{(\ell)}_i$ is stable and represents a clear "forward-in-time" direction for language ℓ at the i-th layer.

Linear structure of time. We test whether calendar values (e.g., a sequence of years {2000, . . ., 2010}) form an underlying linear structure in a 1D subspace of the embedding space. For year calendar component $c \in \{Y\}$ (Year), we train a linear regressor that decodes the corresponding scalar value from the hidden representation $\bar{\mathbf{h}}^{(\ell)}_{y,i}$. Concretely, we fit

$$\hat{c} = \mathbf{W}_c \bar{\mathbf{h}}^{(\ell)}_{y,i} + \mathbf{b}_c, \qquad \text{Linearity}(c) = R^2(c, \hat{c}), \quad (5)$$

where R^2 measures how well the Year values can be recovered by a single linear readout. A higher R^2 indicates that the Year component is organised along an approximately ordered axis in the embedding space, which may help LLMs perform date arithmetic more effectively. We also apply this idea to Month and Day components.

5 Experiments

We examine a diverse suite of decoder-only LLMs to disentangle the effects of model size, architecture, and tokeniser composition. Our open

LLMs include the Qwen3 family (spanning from 0.6B to 14B parameters; (Yang et al., 2025)), LLaMA 3 (8B, 70B; (Touvron et al., 2023)), and variants of OLMo (OLMo et al., 2025; Groeneveld et al., 2024), Gemma (Team et al., 2025), Mistral (Mistral-AI et al., 2025), and Phi-4 (Microsoft et al., 2025). To benchmark against proprietary systems, we evaluate `GPT-4o` and `GPT-4o-mini` (OpenAI et al., 2024). This selection allows us to isolate tokeniser-induced errors (Singh and Strouse, 2024; Lundin et al., 2025b; Bhatia et al., 2025a,b) from obscure failures in temporal reasoning.

5.1 Tokenisation Setup

Baseline vs. model tokenisers. We contrast each model's native subword segmentation against a deterministic, linguistically informed *baseline tokeniser*. This baseline segments date strings into semantic primitives (year, month, day, calendar marker), strictly preserving delimiters and whitespace. It is designed to be language-aware, correctly parsing Arabic-Indic numerals, Chinese temporal markers (e.g., 年, 月), and Hijri suffixes. For each instance in MULTITEMPBENCH, we compute the divergence between the model's native segmentation (using TikToken or Hugging Face tokenizers) and this semantic baseline.

Multilingual date fragmentation ratio (mDFR). We evaluate tokenisation quality using our **mDFR** metric. The learned coefficients for the metric are $\alpha = (0.2, 0.2, 0.1, 0.5)$, (Table 7) reflecting that structural divergence (θ), and root splitting are more detrimental than simple token count inflation. Table 2 provides a qualitative comparison of tokenisation behaviours using the Gemma 3 tokeniser. High-resource languages like German and English exhibit moderate fragmentation (mDFR $\approx$ 0.50–0.53), typically characterised by the splitting of numeric roots (e.g., 2034" becoming `2|0|3|4`) while largely preserving semantic delimiters and month names. In contrast, low-resource settings suffer from semantic fragmentation; for instance, the Hausa date "Oktoba 10, 2034" yields the highest DFR of 0.78, as the month name is broken into opaque sub-word units (`O|kt|oba`) alongside the numeric splitting.

5.2 Temporal Reasoning Evaluation Setup

Prompting strategy. We evaluate models in a zero-shot setting without fine-tuning, chain-of-thought demonstrations, or external knowledge, as these may help LLMs resolve temporal tasks even if date strings are poorly tokenised. Each prompt consists of the question and a concise instruction to output the final answer.

LLM-as-a-judge. Given the diverse output formats across languages, we employ an LLM-based evaluation pipeline. For every prediction, we generate a JSON record containing the question, the model's raw output, and a set of gold-standard aliases (e.g., "03/04/2025", "3 April 2025", ٣ أبريل ٥٢٠٢). `GPT-4o` acts as the judge, classifying the response as `CORRECT` (consistent with gold aliases), `INCORRECT` (mutually exclusive), or `NOT_ATTEMPTED`, which was initially introduced by OpenAI for the QA task (Wei et al., 2024). The automated judge achieved a 87% agreement rate with the majority human vote (Inter annotator agreement Cohen's $\kappa = 0.89$) on a validation set of 250 multilingual instances. (For more details, please see Appendix A.4)

6 Results

Our goal is to address the core question: *what controls temporal reasoning performance, surface tokenisation of dates, or internal geometric structures of temporal representations*? To do so, we first test whether **date fragmentation** predicts accuracy across models and settings (Section 6.2). We then test whether **calendar geometry** in hidden states predicts accuracy (Section 6.3). Finally, we synthesise which factor is necessary and/or sufficient for strong temporal reasoning (Section 6.4).

6.1 Multilingual Temporal Reasoning Performance

Table 3 reports temporal reasoning accuracy averaged across the three tasks within each language. Two patterns stand out. First, performance is highly language-dependent: most models are relatively strong in high-resource languages (English, Chinese, German) but degrade sharply in Hausa, indicating a distinct low-resource regime where temporal reasoning is brittle. Second, model ranking is not explained by model size alone: some smaller open-source LLMs outperform larger ones (for example, the 4B-parameter Gemma 3 achieves a 59.2% average, surpassing both the 8B-parameter Llama 3.1 at 57.3% and the 20B-parameter GPT-OSS at 20.0%), suggesting that multilingual coverage and training/tokenisation choices outrank raw parameter count for this benchmark. These accuracy trends motivate the mechanistic split we

Format	Language	Calendar	Original String	Baseline Tokenization	Gemma 3 Tokenization (Visualized)	mDFR
DD. Month YYYY	German	Greg.	10. Oktober 2034	10 | . | Oktober | 2034	1 | 0 | . | Oktober | 2 | 0 | 3 | 4	0.50
Month DD, YYYY	English	Greg.	October 10, 2034	October | 10 | , | 2034	October | 1 | 0 | , | 2 | 0 | 3 | 4	0.53
YYYY年MM月DD日	Chinese	Greg.	2034 年10 月10 日	2034 | 年 | 10 | 月 | 10 | 日	2 | 0 | 3 | 4 | 年 | 1 | 0 | 月 | 1 | 0 | 日	0.55
DD Month YYYY هـ	Arabic	Hijri	٧٢ رجب ٦٥٤١ هـ	٧٢ | رجب | ٦٥٤١ | هـ	٢ | ٧ | ر | جب | ١ | ٤ | ٥ | ٦ | هـ	0.60
DD Month YYYY AH	English	Hijri	27 Rajab 1456 AH	27 | Rajab | 1456 | AH	2 | 7 | Raj | ab | 1 | 4 | 5 | 6 | AH	0.60
干支年 MM月DD	Chinese	Lunar	辛亥年五月廿三	辛亥 | 年 | 五月 | 廿三	辛亥 | 年 | 五 | 月 | 廿三	0.65
DD Month YYYY	Arabic	Greg.	٠١ أكتوبر ٤٣٠٢	٠١ | أكتوبر | ٤٣٠٢	١ | ٠ | أكتوبر | ٢ | ٠ | ٣ | ٤	0.70
DD Month YYYY	English	Greg.	10 October 2034	10 | October | 2034	1 | 0 | October | 2 | 0 | 3 | 4	0.75
Month DD, YYYY	Hausa	Greg.	Oktoba 10, 2034	Oktoba | 10 | , | 2034	O | kt | oba | 1 | 0 | , | 2 | 0 | 3 | 4	0.78

Table 2: **Qualitative Analysis of Tokenisation Fragmentation.** Vertical bars (|) denote token boundaries within the Gemma 3 tokenizer. The semantic baseline is a human-interpretable segmentation of date units and delimiters.

	Accuracy by Language (%)					
Model	Arabic	Chinese	English	German	Hausa	Average
Proprietary Models						
GPT-4o	**71.3**	**66.0**	54.3	**70.0**	**51.7**	**62.7**
Open-Weights Models						
Gemma 3 4B	57.3	64.7	63.7	64.0	46.3	**59.2**
Llama 3.1 8B	49.0	65.0	**66.7**	64.3	41.3	57.3
Phi-4 Mini	39.7	55.7	66.3	62.7	28.7	50.6
Qwen 3 4B	41.3	56.7	54.7	46.0	9.0	41.5
Mistral 7B v0.2	44.3	40.7	51.7	54.7	9.0	40.1
Llama 2 7B	15.7	40.7	55.0	51.3	17.0	35.9
Gemma 3 1B	27.3	42.7	40.7	38.0	19.3	33.6
Olmo 3 7B	16.3	39.3	48.0	33.3	12.3	29.8
DS-R1 Qwen 7B	24.7	48.0	45.3	42.3	1.7	32.4
OLMo 2 7B	16.3	39.3	48.0	33.3	12.3	29.9
GPT-OSS 20B	5.0	24.0	49.0	20.3	2.0	20.0
Qwen 3 14B	25.3	9.7	19.0	27.7	2.3	16.8
Qwen3 0.6B	21.7	19.7	14.7	23.3	4.0	16.6

Table 3: **Multilingual Temporal Reasoning Accuracy.** Accuracy is averaged across the three tasks (date arithmetic, time zone conversion, temporal relation extraction) within each language.

test next: low-resource failures align with an *input accessibility* bottleneck (date fragmentation), whereas high-resource variation is better explained by an *internal geometry* bottleneck (temporal linearity).

Calendar	Gregorian					Lunar	Hijri		
Model	Ar	Zh	En	De	Ha	Zh	Ar	En	Ha
Baseline	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GPT-3.5	0.19	0.12	0.23	0.12	0.12	0.41	0.14	0.31	0.30
GPT-4	0.19	0.12	0.23	0.12	0.12	0.41	0.14	0.31	0.30
GPT-5	0.19	0.12	0.23	0.12	0.12	0.41	0.14	0.31	0.30
Llama 2	0.06	0.23	0.42	0.30	0.29	0.20	0.04	0.23	0.23
Phi 3.5	0.06	0.23	0.42	0.30	0.29	0.20	0.04	0.23	0.23
Mistral	0.06	0.23	0.42	0.30	0.29	0.29	0.04	0.23	0.23
Davinci-003	0.17	0.17	0.37	0.10	0.16	0.29	0.09	0.37	0.39
OLMo	0.13	0.18	0.37	0.09	0.16	0.41	0.09	0.37	0.40
Llama 3	0.35	0.16	0.34	0.12	0.12	0.60	0.36	0.31	0.30
DeepSeek	0.10	0.31	0.44	0.34	0.32	0.61	0.05	0.30	0.29
gpt-oss	0.39	0.16	0.34	0.12	0.13	0.60	0.44	0.32	0.32
Qwen3	0.17	0.32	0.44	0.34	0.32	0.61	0.16	0.31	0.28
Cohere	0.18	0.34	0.44	0.34	0.33	0.63	0.22	0.32	0.32
Gemma3	0.39	0.34	0.44	0.34	0.33	0.48	0.30	0.31	0.31

Table 4: Multilingual Date Fragmentation Ratio (mDFR) across models for Gregorian (Ar, Zh, En, De, Ha), Lunar (Zh), and Hijri (Ar, En, Ha). Higher scores indicate greater fragmentation of date tokens.

6.2 Date Fragments and Temporal Reasoning

Fragmentation varies by language and calendar format. Table 4 reports mDFR across models and calendar varieties. Fragmentation arises from digit splitting (e.g., 2034 → 2|0|3|4)(Table 2). These effects are amplified in low-resource settings and in less frequent calendar variants: for instance, non-Gregorian formats are often incorrectly tokenised, with more date fragments than Gregorian formats (Table 2), due to sparse calendar markers and lower-frequency month lexemes.

Date fragments are a major bottleneck for temporal reasoning in low-resource languages. Figure 2 shows that the greater the mDFR, the lower the accuracy in the date arithmetic task for two low-resource languages: Hausa ($r = -0.97$) and Arabic ($r = -0.89$). However, the correlation becomes much weaker for the three high-resource languages: German, Chinese, and English, indicating that more date fragments do not cause an accuracy collapse for these languages. Overall, these results suggest that tokenisation is a major bottleneck for temporal reasoning in low-resource languages, but not in high-resource ones: Bhatia et al. (2025a) explained why this is the case in English: they found that LLMs can compensate for date fragmentation by stitching fragmented date tokens during temporal reasoning. We speculate that in high-resource languages, such fragments are still frequently observed in training data, which enables LLMs, especially larger models, to internally address/stitch them. We observe similar findings in the other two tasks: time zone conversion and temporal relation (see Figure 6 and Figure 7).

6.3 Geometric Structures and Temporal Reasoning

Tokenisation characterises what information is presented to the model, but not whether the model

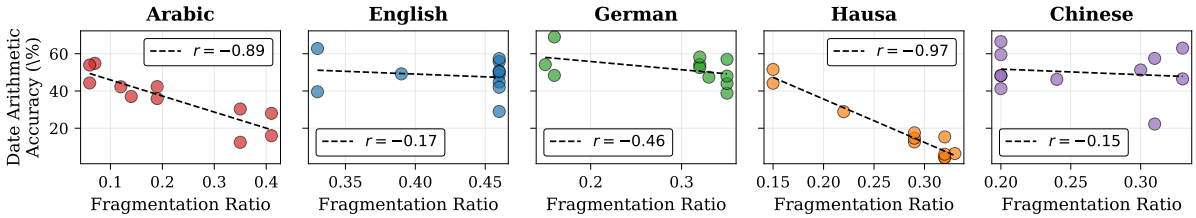


Figure 2: **Impact of Tokenisation on Date Arithmetic Accuracy.** DFR is strongly negatively correlated with accuracy in Hausa ($r = -0.97$), but only weakly correlated in English ($r = -0.17$).

organises that information in a form suitable for computation. We therefore test whether temporal representations possess internal geometric structures that support calendar manipulation. Here we focus on **temporal linearity**: how well temporal values (e.g., years) are organised along approximately ordered 1D axes, measured by the R^2 of linear probes decoding *Day*, *Month*, and *Year* from hidden states.

Temporal linearity is a strong predictor of performance in high-resource languages. Figure 3 plots the average temporal reasoning accuracy (across date arithmetic, time zone conversion, and temporal relation extraction) against overall temporal linearity (aggregated across all calendar components, including delimiters) across models within each language. Overall, temporal linearity is strongly correlated with accuracy in English ($r = 0.77$) and Chinese ($r = 0.75$), moderately correlated in German ($r = 0.44$) and Arabic ($r = 0.34$), and weakly correlated in Hausa ($r = 0.10$). This pattern suggests that, once models can reliably access calendar components, strong performance depends on representing temporal *values* in an ordered geometry that supports arithmetic-like updates. In other words, in high-resource settings the main limiter is not surface form, but whether the model embeds time on a usable internal axis.

Component-wise view: the Year axis is typically the most predictive. While Figure 3 summarises the holistic relationship between temporal linearity and accuracy, noting that this overall correlation is distinct from a simple average of the individual components, Figure 4 decomposes this geometry by calendar component (Day/Month/Year) within each language. Two trends stand out. First, correlations are generally strongest for **Year** (especially in English/Chinese), consistent with year values providing the primary ordered backbone required for many temporal operations. Second, **Month** and **Day** linearity show weaker and more heterogeneous correlations across languages. This suggests that Month and Day representations are not as robustly formed as Year representations, and are instead more sensitive to language- and format-specific cues (e.g., month lexemes and delimiters) than to a universal ordered axis. Overall, the component-wise breakdown supports the interpretation that ordered temporal geometry matters most when it provides a stable *year* backbone, and that this signal is clearest in high-resource languages.

6.4 Which Mechanism Controls Temporal Reasoning?

To test the relative contribution of **date fragmentation** and **temporal linearity**, we fit a crossed mixed-effects regression predicting **per-question accuracy** over all 285000 predictions (15000 questions $\times$ 19 models). The dependent variable was binary correctness for each prediction. As fixed effects, we included z-scored **mDFR**, z-scored **linearity**, **resource level** (high-resource vs. low-resource), and all interaction terms: $\texttt{correct} \sim \texttt{mDFR}_z * \texttt{linearity}_z * \texttt{resource}$. We also included crossed random intercepts for **question** and **model** to account for item difficulty and model-specific baseline performance. This analysis lets us test whether temporal reasoning performance is better explained by **surface tokenisation of dates** or by **internal geometric structures of temporal representations**, and whether this differs by resource level. The regression confirms that temporal reasoning performance is governed by a **language-dependent bottleneck**. We report regression coefficients using the notation β (coefficient), SE (standard error), z (Wald statistic), and p (p-value). Most importantly, the three-way interaction between **mDFR**, **linearity**, and **resource level** is significant ($\beta = 0.016$, $SE = 0.007$, $z = 2.31$,

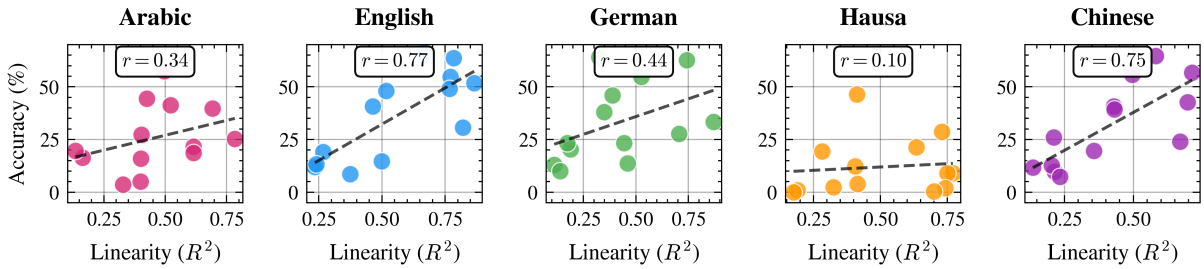


Figure 3: **Temporal linearity vs. accuracy across languages.** Temporal linearity (probe R^2) is strongly correlated with accuracy in English (r=0.77) and Chinese (r=0.75), but weakly correlated in Hausa (r=0.10), suggesting that ordered temporal geometry is a key driver of high performance when it emerges.

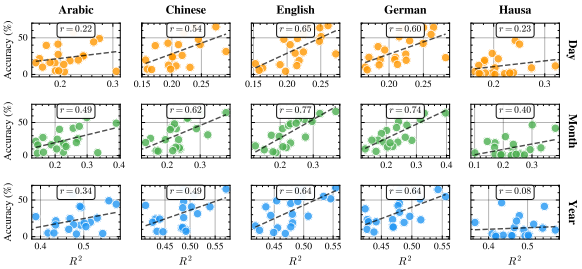


Figure 4: **Component-wise temporal linearity vs. accuracy.** Correlations between accuracy and probe R^2 for **Day**, **Month**, and **Year** within each language.

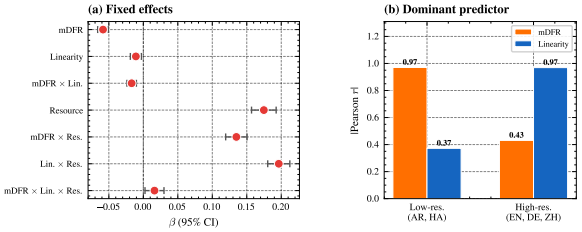


Figure 5: **Mixed-effects summary of temporal reasoning bottlenecks.** (a) Fixed effects from the crossed mixed-effects regression. (b) Dominant predictor by resource regime: mDFR in low-resource languages, linearity in high-resource languages.

$p = 0.021$), showing that the dominant predictor changes across language regimes. In **low-resource** languages (Arabic and Hausa), higher fragmentation strongly predicts lower accuracy ($\beta = -0.126$, $p < 0.001$), indicating that **date fragmentation** is the dominant bottleneck. In **high-resource** languages (English, German, and Chinese), **temporal linearity** is instead the stronger predictor of accuracy ($\beta = 0.087$, $p < 0.001$), while mDFR has only a weak effect ($\beta = 0.009$, $p = 0.056$). These results align with the language-wise analyses above: **low-resource languages tend to be input-limited**, whereas **high-resource languages tend to be geometry-limited**. Figure 5a summarises these results across the MULTITEMPBENCH. Figure 5a also shows that the main effects of resource level and its interactions with mDFR and linearity are the largest fixed effects in the model, consistent with a resource-dependent shift in the dominant bottleneck. As shown in Figure 5b, this yields a clear split: **low-resource languages tend to be input-limited**, whereas **high-resource languages tend to be geometry-limited**. Overall, no single factor universally predicts temporal reasoning across languages; instead, the dominant constraint shifts from **date fragmentation** to **temporal linearity** as resource level increases. Mixed-effects regression goes beyond pairwise correlation by controlling for grouped variation across questions and models, but it remains observational rather than causal. Practically, our results suggest different mitigation paths: date normalisation or tokeniser redesign for low-resource settings, and temporal-representation alignment for high-resource settings.

7 Conclusion

MULTITEMPBENCH shows that multilingual temporal intelligence depends on more than adding vocabulary: it requires making temporal information *accessible* and *computable* in the model's internal space. A crossed mixed-effects regression confirms this language-dependent bottleneck: in low-resource regimes, date fragmentation is the stronger predictor of failure, while in high-resource regimes temporal linearity is the stronger predictor of performance. This points to actionable mitigation paths: improving date access through normalisation/tokeniser design in low-resource settings, and improving internal temporal geometry through representation-level alignment in high-resource settings.

Limitations

MULTITEMPBENCH is designed as a controlled diagnostic, and that design imposes constraints on generality: it covers five languages (English, German, Chinese, Arabic, and Hausa) and three task families (date arithmetic, time zone conversion, and temporal relation extraction), so it does not fully represent the diversity of multilingual temporal phenomena (e.g., additional scripts and dialects, code-mixing/noisy text, domain-specific jargon, or other calendar conventions beyond those included); instances are produced via translation and templated format variation from a curated English seed set, which helps isolate surface-form/tokenisation effects but may under-sample naturally occurring distributions of expressions and errors; we evaluate in a zero-shot, direct-answer setting (and normalise outputs with an LLM-as-a-judge), which improves comparability yet may understate performance under tool use, prompting, or fine-tuning and introduces residual evaluation noise from judge mistakes and format ambiguity; and while we find strong associations between fragmentation metrics, temporal linearity, and performance, our mechanistic analyses remain correlational and probe-centric, leaving open causal questions about tokeniser design, training data, decoding dynamics, and non-linear representational structure. A key limitation of our multilingual setup is that the low-resource regime is represented by only two languages (Arabic and Hausa), so claims about low-resource temporal reasoning should be interpreted as suggestive rather than fully general across low-resource languages. More broadly, the language split into high-resource versus low-resource is necessarily coarse, and additional languages are needed to test whether the same bottleneck pattern holds across other typological profiles, scripts, and calendar traditions.

Ethical Considerations

This benchmark surfaces disparities that can arise from multilingual tokenisation and resource imbalance, but such results should be framed as properties of model design and data coverage rather than as inherent deficits of particular languages to avoid reinforcing harmful narratives; because calendar expressions are culturally situated (including non-Gregorian systems such as Hijri and Chinese Lunar), conversion or formatting errors can have real consequences in downstream, potentially high-stakes contexts, so users should document conversion assumptions and validate systems with native-speaker and domain-expert review when decisions matter; the dataset is constructed from public sources through translation and controlled transformations and is not intended to contain personal data, yet extensions should avoid introducing identifiable or sensitive information and, the human annotation is used follows informed consent and fair compensation practices; finally, any dependence on third-party model APIs for translation and/or evaluation can affect reproducibility and raise governance concerns, so releases will document versions and settings and, where feasible, provide open alternatives, while acknowledging that improved temporal reasoning can be dual-use and warrants domain-specific risk assessment and human oversight in sensitive deployments.

Broader Impact

By providing a controlled multilingual temporal benchmark and analysis signals (e.g., fragmentation and representation geometry probes), this work can help the community audit and improve temporal reasoning across scripts, languages, and calendar conventions, potentially reducing "token tax" effects and improving language equity in multilingual NLP; it may also guide more principled tokeniser and training-data interventions by linking surface segmentation properties to downstream competence; however, like any benchmark, it can distort incentives if treated as a leaderboard target, encouraging optimisation for templated formats or discouraging support for languages that score poorly, so we emphasise its role as a diagnostic instrument rather than a deployment-readiness test and encourage follow-on work to broaden coverage (more languages/dialects and naturalistic temporal text), evaluate mitigation strategies directly, and report results with uncertainty and careful error analysis to support responsible, inclusive progress.

References

Orevaoghene Ahia, David de Almeida, Nathan Shleifer, and Emily Dinan. 2023. Do all languages cost the same? tokenization in the era of commercial language models. *arXiv preprint arXiv:2305.13707*.

Mohammed H Alshehri. 2024. Hijridate: A python package for hijri-gregorian date conversion.

Gagan Bhatia, Maxime Peyrard, and Wei Zhao. 2025a. Date fragments: A hidden bottleneck of tokenization for temporal reasoning. In *Proceedings of the 2025*

Conference on Empirical Methods in Natural Language Processing, pages 3201–3219, Suzhou, China. Association for Computational Linguistics.

Gagan Bhatia, Ming Ze Tang, Cristina Mahanta, Madiha Kazi, Maxime Peyrard, and Wei Zhao. 2025b. DateLogicQA: Benchmarking temporal biases in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 321–332, Albuquerque, USA. Association for Computational Linguistics.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Haotian Wang, Ming Liu, and Bing Qin. 2023. Timebench: A comprehensive evaluation of temporal reasoning abilities in large language models. *arXiv preprint arXiv:2311.17667*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Ahmed Oumar El-Shangiti, Tatsuya Hiraoka, Hilal AlQuabeh, Benjamin Heinzerling, and Kentaro Inui. 2025. The geometry of numerical reasoning: Language models compare numeric properties in linear subspaces. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 550–561, Albuquerque, New Mexico. Association for Computational Linguistics.

Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. 2024. Test of time: A benchmark for evaluating llms on temporal reasoning. *arXiv preprint arXiv:2406.09170*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, and 24 others. 2024. Olmo: Accelerating the science of language models. *Preprint*, arXiv:2402.00838.

Wes Gurnee and Max Tegmark. 2024. Language models represent space and time. In *Proceedings of the 12th International Conference on Learning Representations*. ICLR 2024.

Yicheng Han, Shih-Ming Wang, Jialu Zhang, Qian Liu, and Wei Lu. 2025. Ticktack: Modeling temporal relationships in llms using non-gregorian calendars. *arXiv preprint arXiv:2503.04150*.

Carolin Holtermann, Paul Röttger, and Anne Lauscher. 2025. Around the world in 24 hours: Probing LLM knowledge of time and place. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22875–22897, Vienna, Austria. Association for Computational Linguistics.

Duygu Sezen Islakoglu and Jan-Christoph Kalo. 2025. Chronosense: Exploring temporal understanding in large language models with time intervals of events. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Short Papers)*. ACL 2025.

Vani Kanjirangat, Tanja Samardžić, Ljiljana Dolamic, and Fabio Rinaldi. 2025. Tokenization and representation biases in multilingual models on dialectal NLP tasks. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. EMNLP 2025.

N. J. Karthika, Maharaj Brahma, Rohit Saluja, Ganesh Ramakrishnan, and Maunendra Sankar Desarkar. 2025. Multilingual tokenization through the lens of indian languages: Challenges and insights. *arXiv preprint arXiv:2506.17789*.

Linus Kreitner, Paul Hager, Jonathan Mengedoht, Georgios Kaissis, Daniel Rueckert, and Martin J. Menten. 2025. Efficient numeracy in language models through single-token number embeddings. *arXiv preprint arXiv:2510.06824*.

Aochong Oliver Li and Tanya Goyal. 2025. Memorization vs. reasoning: Updating LLMs with new knowledge. In *Findings of the Association for Computational Linguistics: ACL 2025*. ACL Findings 2025.

Huihan Li, You Chen, Siyuan Wang, Yixin He, Ninareh Mehrabi, Rahul Gupta, and Xiang Ren. 2025. Diagnosing memorization in chain-of-thought reasoning, one token at a time. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. EMNLP 2025.

Zefang Liu, Nam H. Nguyen, Yinzhu Quan, and Shi-Xiong Zhang. 2025. Temporal tokenization strategies for event sequence modeling with large language models. *Preprint*, arXiv:2512.13618.

Jessica M. Lundin, Ada Zhang, Nihal Karim, Hamza Louzan, Victor Wei, David Adelani, and Cody Carroll. 2025a. The token tax: Systematic bias in multilingual tokenization. *arXiv preprint arXiv:2509.05486*.

Jessica M. Lundin, Ada Zhang, Nihal Karim, Hamza Louzan, Victor Wei, David Adelani, and Cody Carroll. 2025b. The token tax: Systematic bias in multilingual tokenization. *Preprint*, arXiv:2509.05486.

Siddarth Mamidanna, Daking Rai, Ziyu Yao, and Yilun Zhou. 2025. All for one: LLMs solve mental math at the last token with information transferred from other tokens. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. EMNLP 2025.

Vittorio Mazzia, Sandro Pollastrini, Davide Bernardi, Chiara Rubagotti, and Daniele Amberti. 2026. Benchmarking multilingual temporal reasoning in

LLMs: The temporal reasoning dataset. In *Proceedings of the 16th International Workshop on Spoken Dialogue System Technology*, pages 168–181, Trento, Italy. Association for Computational Linguistics.

Zeyu Miao and 1 others. 2025a. Benchmarking and improving cross-calendar temporal reasoning of large language models. *Preprint*, arXiv:2511.09993.

Zhongjian Miao, Hao Fu, and Chen Wei. 2025b. Span: Benchmarking and improving cross-calendar temporal reasoning of large language models. *Preprint*, arXiv:2511.09993.

Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, and 57 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *Preprint*, arXiv:2503.01743.

Mistral-AI, :, Abhinav Rastogi, Albert Q. Jiang, Andy Lo, Gabrielle Berrada, Guillaume Lample, Jason Rute, Joep Barmentlo, Karmesh Yadav, Kartik Khandelwal, Khyathi Raghavi Chandu, Léonard Blier, Lucile Saulnier, Matthieu Dinot, Maxime Darrin, Neha Gupta, Roman Soletskyi, Sagar Vaze, and 82 others. 2025. Magistral. *Preprint*, arXiv:2506.10910.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, and 24 others. 2025. 2 olmo 2 furious. *Preprint*, arXiv:2501.00656.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. 2025. Fineweb2: One pipeline to scale them all–adapting pre-training data processing to every language. *arXiv preprint arXiv:2506.20920*.

Aleksandar Petrov, Emanuele La Malfa, Philip HS Torr, and Adel Bibi. 2023. Language model tokenizers introduce unfairness between languages. *arXiv preprint arXiv:2305.15425*.

Piotr Pęzik, Konrad Kaczyński, Maria Szymańska, Filip Żarnecki, Zuzanna Deckert, Jakub Kwiatkowski, and Wojciech Janowski. 2025. Llmlagbench: Identifying temporal training boundaries in large language models. *Preprint*, arXiv:2511.12116.

Mutsumi Sasaki, Go Kamoda, Ryosuke Takahashi, Kosuke Sato, Kentaro Inui, Keisuke Sakaguchi, and Benjamin Heinzerling. 2025. Can language models handle a non-gregorian calendar? the case of the Japanese wareki. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 444–463, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Rohit Saxena, Aryo Pradipta Gema, and Pasquale Minervini. 2025. Lost in time: Clock and calendar understanding challenges in multimodal LLMs. *arXiv preprint arXiv:2502.05092*.

Aaditya K. Singh and D. J. Strouse. 2024. Tokenization counts: The impact of tokenization on arithmetic in frontier LLMs. *arXiv preprint arXiv:2402.14903*.

Dimitris Spathis and Fahim Kawsar. 2023. The first step is the hardest: Pitfalls of representing and tokenizing temporal data for large language models. *arXiv preprint arXiv:2309.06236*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

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

Yuqing Wang and Yun Zhao. 2024. Tram: Benchmarking temporal reasoning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*.

Zhengxiang Wang and Zeyu Dong. 2026. Measuring iterative temporal reasoning with time puzzles. *Preprint*, arXiv:2601.07148.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*.

Shaohang Wei, Wei Li, Feifan Song, Wen Luo, Tianyi Zhuang, Haochen Tan, Zhijiang Guo, and Houfeng Wang. 2025. Time: A multi-level benchmark for temporal reasoning of LLMs in real-world scenarios. In *Advances in Neural Information Processing Systems*. NeurIPS 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41

others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Chenghao Zhu, Nuo Chen, Yufei Gao, Yunyi Zhang, Prayag Tiwari, and Benyou Wang. 2024. Evaluating LLMs at Evaluating Temporal Generalization. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers).*

A Appendix

A.1 Creation of our MultiTempBench

To ensure consistency across the multilingual benchmark, we implemented a unified processing pipeline. This pipeline processes the English source data and generates language-specific variants for Arabic, Chinese, Hausa, German, and English. The process consists of two stages: *Standardization* and *Polymorphic Formatting*.

A.1.1 Stage 1: Date Extraction and Standardization

The first step is identical for all five languages. We utilize a regular expression to identify date entities within the source text. Regardless of the input format, these dates are parsed into a standard internal representation (Year, Month, Day). This ensures that all downstream formatters operate on a consistent temporal grounding.

A.1.2 Stage 2: Polymorphic Formatting

Once standardized, the pipeline applies four distinct formatters per language: **ISO**, **Slash** (Numeric), **Long** (Textual), and **Calendar** (Phrasal/Cultural). The specific logic for each language is detailed below. The conversion process was implemented through a unified Python pipeline. For each language, the system first extracts and parses dates from the source English questions into a standard internal representation. Language-specific formatters are then applied. For instance, Arabic formatting involves converting digits to Arabic-Indic numerals, applying right-to-left marks for ISO dates, and using the hijri-converter library to generate Hijri calendar dates (e.g., ذو الحجة ١٤٤١هـ). Similarly, Chinese formatting integrates conversions to the traditional lunar calendar.

Arabic Implementation. The Arabic formatting pipeline requires specific handling for text directionality and numeral systems.

- **ISO Format:** To prevent rendering issues in Right-to-Left (RTL) contexts, the standard ISO string is wrapped in Unicode Left-to-Right Marks (LRM, `U+200E`).
- **Long Format:** We map Gregorian month indices to their Arabic counterparts (e.g., *July* → *Yuliyu*) and convert Western Arabic numerals (0-9) to Eastern Arabic-Indic numerals (٠ ٩).
- **Calendar (Hijri) Format:** We utilize the `hijri-converter` library to transform the Gregorian date into the Hijri calendar. The resulting day, month, and year are formatted using standard Hijri month names (e.g., *Ramadan*, *Shawwal*).

Chinese Implementation. Chinese formatting emphasizes the use of component suffixes and Lunar conversion.

- **Long Format:** Adheres to the standard East Asian order (Year-Month-Day) with the respective character suffixes (年, 月, 日).
- **Calendar (Lunar) Format:** We convert the Gregorian date to the Chinese Lunar calendar using the `lunarcalendar` library. The numeric years are converted to their Chinese character equivalents (e.g., 2023 → 二零二三), and months are mapped to their traditional lunar representations.

Hausa Implementation. Hausa formatting integrates Islamic cultural elements with standard Gregorian tracking, reflecting the region's dual-calendar usage.

- **Long Format:** Uses the particle "ga" (meaning "on") to connect the day and the month (e.g., *03 ga Afrilu 2023*).
- **Calendar Format:** In this variant, we utilize the locally recognized Islamic month names (e.g., *Ramadan*, *Shawwal*) while maintaining the Gregorian year for clarity in civil contexts.

German Implementation. German requires specific grammatical phrasings for the "Calendar" variant to represent a formal date expression. While the standard formats use dot separators (DD.MM.YYYY), the calendar variant expands this to a formal phrase: "Am [Day]. [Month] des Jahres [Year]" (e.g., *Am 26. Juni des Jahres 2025*).

Task	Raw	Fmt	Lng	Size	Example	GT
Arithmetic	250	4	5	5,000	In a movie, the tower took exactly 14 years to construct. They started in 2000-12-27. When was it ready?	2014-12-27
Time Zone	250	4	5	5,000	If it's 2 AM on 1352-03-02 in Asia/Singapore, what's the date and time in Europe/Athens?	8 PM on 1352-03-01
Relation	250	4	5	5,000	Rules for lending against stocks and unit trusts were also redefined. What is the relationship between the event 'redefined' and the time 'April 1, 1997'?	IS_INCLUDED
Total	**750**	**6**	**5**	**15,000**		

Table 5: **Overview of tasks in the MULTITEMPBENCH dataset.** "Raw" denotes unique English questions. "Size" is the total number of examples after multilingual/format expansion ($Raw \times 4$ Fmt $\times$ 5 Lang). The Truth column shows the expected answer format. We have 6 unique date formats.

A.1.3 Examples

Example of our MULTITEMPBENCH is provided in Table 5.

A.2 Validation of Multilingual Date Fragmentation Ratio (mDFR)

This appendix provides a detailed account of the formulation and two-part validation process for our custom Multilingual Date Fragmentation Ratio (mDFR). We demonstrate that this metric aligns closely with human intuition regarding semantic disruption and relies on empirically sound weightings.

A.2.1 Metric Formulation

We calculate the structural divergence θ between the model's token count vector $\mathbf{t}$ and the semantic baseline vector $\mathbf{b}$ using cosine distance. This metric quantifies the deviation of the model's tokenisation from an ideal semantic segmentation. The final mDFR score, $F \in [0, 1]$, is constructed as a weighted sum of four specific error components: whether semantic roots are split ($\mathbb{1}_{\text{split}}$), whether delimiters are lost ($\mathbb{1}_{\text{delimiter}}$), the increase in total token count (ΔN), and the distributional divergence (θ).

A.2.2 Human Evaluation of Fragmentation Severity

This study was designed to confirm that our F metric captures what humans perceive as semantic disruption in tokenized dates more effectively than general-purpose text similarity metrics.

Methodology. We recruited five computer science graduate students, who were familiar with NLP but blind to our hypotheses, to serve as annotators. We created a stimulus set of 100 tokenised date strings, stratified to represent a wide range of models, date formats, and fragmentation levels from our experiments. For each item, annotators were shown the original date and the list of sub-tokens, and asked to rate the **"fragmentation severity"** on a 5-point Likert scale, according to the following rubric:

- **1 (No Fragmentation):** Tokens perfectly preserve the semantic components.
- **2 (Minor Fragmentation):** Mostly preserved, with minor, non-ideal splits.
- **3 (Moderate Fragmentation):** Core components are broken, making the structure harder to discern. Delimiters might be lost or numbers oddly grouped.
- **4 (High Fragmentation):** Date split into many small pieces (e.g., single digits), though the original characters are easily reassembled.
- **5 (Severe Fragmentation):** tokenisation completely obscures the date's structure, often by adding non-numeric tokens or creating highly unintuitive groupings.

The human judgments were highly reliable, with a Krippendorff's Alpha for inter-annotator agreement of $\alpha = 0.81$.

Results. We computed the Spearman's rank correlation coefficient (ρ) between the average human rating for each item and the scores from our F metric, BLEU, and character-level Edit Distance. As shown in Table 6, our F metric demonstrated a strong correlation with human ratings ($\rho = 0.89$), far exceeding general-purpose metrics like BLEU ($\rho = 0.43$).

A.2.3 Data-Driven Validation of Metric Coefficients

To directly tune our metric to align with human perception, we framed the weight determination

Metric	Correlation (ρ)
mDFR	**0.89**
DFR (Bhatia et al., 2025a)	<u>0.81</u>
BLEU Score	0.43
Character-Level Edit Distance	0.29

Table 6: Spearman Correlation (ρ) of Metrics with Human Judgments of Fragmentation Severity for Multilingual dates.

as a linear regression problem. The goal was to predict the average human severity rating using the four fragmentation components as features: $\mathbf{x} = [\mathbb{1}_{\text{split}}, \mathbb{1}_{\text{delimiter}}, (N - N_b), \theta]$.

After fitting the model to our human evaluation data, we obtained a set of empirically derived coefficients. As shown in Table 7, the weights learned from human ratings are remarkably similar to the normalised version of our original, intuitively set weights. This confirms that Distributional Divergence (θ) is the dominant factor in perceived severity, followed by structural breaks, with token count inflation playing a minor role.

A.2.4 Qualitative Analysis of Fragmentation

To visualise how mDFR scores correspond to real-world model outputs, we analysed tokenisation patterns across different languages and scripts. Table 2 illustrates the correlation between high mDFR scores, human severity ratings, and severe segmentation issues. Notably, non-Latin scripts (e.g., Arabic, Chinese) and agglutinative languages often suffer from higher fragmentation (rated 4.6–5.0 by humans), where semantic roots are often shattered into single characters or bytes.

A.3 Correlation of the different tasks

The same broad pattern holds in the other two tasks, though the strength of the effect varies by task. In **temporal relation extraction** (Figure 6), higher fragmentation is associated with lower accuracy in the two low-resource languages, especially Hausa ($r = -0.58$), while the relationship remains weak or near-zero in English ($r = 0.06$), German ($r = 0.28$), and Chinese ($r = 0.08$). Arabic also shows a modest negative correlation ($r = -0.29$). As in date arithmetic, these results suggest that **date fragmentation** is more consequential in low-resource settings, whereas high-resource languages are generally more robust to fragmented temporal inputs.

A similar but slightly stronger pattern appears in **time zone conversion** (Figure 7). Fragmentation is negatively correlated with accuracy in Arabic ($r = -0.54$) and especially Hausa ($r = -0.74$), but remains weak in English ($r = -0.15$), German ($r = -0.01$), and Chinese ($r = -0.13$). Compared with temporal relation extraction, time zone conversion shows a clearer low-resource penalty, though still less extreme than the effect observed for date arithmetic. Overall, across all three tasks, **date fragmentation** is most predictive of failure in low-resource languages, supporting the view that tokenisation is a regime-dependent bottleneck rather than a universal explanation of temporal reasoning errors.

A.4 Human Evaluation Details.

To validate the reliability of the LLM-based judging pipeline, we conducted a human evaluation on a subset of the benchmark. Six annotators participated in the study, all of whom were Master's students in computer science or closely related disciplines. For each language included in the validation set, at least two annotators independently reviewed the model outputs and determined whether the response should be classified as `CORRECT`, `INCORRECT`, or `NOT_ATTEMPTED`. The evaluation covered multiple languages present in the benchmark to ensure that linguistic diversity did not bias the assessment. Disagreements were resolved using majority voting across annotators. Across the evaluated instances, the human annotators achieved an average agreement rate of approximately 89%, indicating strong consistency in the evaluation criteria. This agreement level provides additional confidence in the reliability of the automated LLM-as-a-judge evaluation protocol used in our experiments.

A.5 Temporal Geometry

PCA visualization across layers. To provide a qualitative view of how temporal structure emerges across depth, we apply PCA to the set of points $\{\bar{\mathbf{h}}_{y,i}^{(\ell)}\}$ for $y \in [1990, 2024]$ across the five languages. In the visualizations (Figure 8), the resulting plots display the sequence of line segments connecting consecutive years, revealing whether languages form coherent, linear paths in the embedding space.

Fragmentation Component	Original Intuitive Weight (Normalised)	Empirically Learned Weight (from Human Ratings)
1_{split} (Component Split)	0.1818	**0.2015**
$1_{\text{delimiter}}$ (Delimiter Loss)	0.1818	**0.1932**
$N - N_b$ (Token Difference)	0.0909	**0.1053**
θ (Distributional Divergence)	0.5455	**0.5000**

Table 7: Comparison of Original (Normalised) and Empirically Learned Weights for the F Metric.

Format	Language	Calendar	Original String	Baseline tokenisation	Gemma 3 tokenisation (Visualized)	mDFR	Avg. Human Rating
DD. Month YYYY	German	Greg.	10. Oktober 2034	10 . Oktober 2034	1 | 0 | . | Oktober | 2 | 0 | 3 | 4	0.50	4.2
Month DD, YYYY	English	Greg.	October 10, 2034	October 10 , 2034	October | 1 | 0 | , | 2 | 0 | 3 | 4	0.53	4.4
YYYY年MM月DD日	Chinese	Greg.	2034 年10 月10 日	2034 年10 月10 日	2 | 0 | 3 | 4 | 年 | 1 | 0 | 月 | 1 | 0 | 日	0.55	4.6
DD Month YYYY هـ	Arabic	Hijri	٧٢ رجب ٦٥٤١ هـ	٧٢ رجب ٦٥٤١ هـ	٢ | ٧ | ر | جب | ١ | ٤ | ٥ | ٦ | هـ	0.60	5.0
DD Month YYYY AH	English	Hijri	27 Rajab 1456 AH	27 Rajab 1456 AH	2 | 7 | Raj | ab | 1 | 4 | 5 | 6 | AH	0.60	5.0
DD Month YYYY	Arabic	Greg.	٠١ أكتوبر ٤٣٠٢	٠١ أكتوبر ٤٣٠٢	١ | ٠ | أكتوبر | ٢ | ٠ | ٣ | ٤	0.70	5.0
DD Month YYYY	English	Greg.	10 October 2034	10 October 2034	1 | 0 | October | 2 | 0 | 3 | 4	0.75	5.0
Month DD, YYYY	Hausa	Greg.	Oktoba 10, 2034	Oktoba 10 , 2034	O | kt | oba | 1 | 0 | , | 2 | 0 | 3 | 4	0.78	5.0

Table 8: **Qualitative Analysis of Tokenisation Fragmentation.** Vertical bars (|) denote token boundaries within the Gemma 3 tokeniser. The **Avg. Human Rating** (1-5 scale) confirms that higher mDFR scores correspond to perceived severe fragmentation.

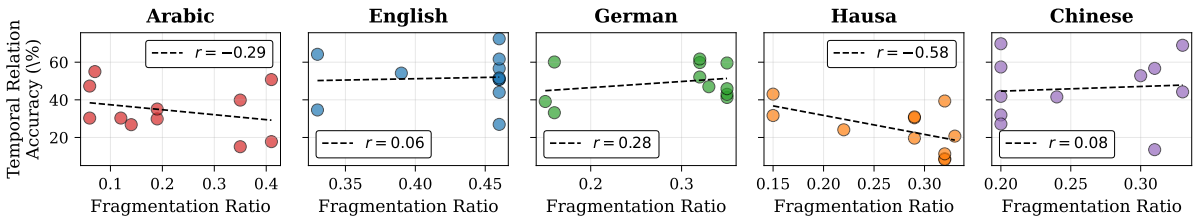


Figure 6: **Impact of Tokenisation on Temporal Relation Task Accuracy.** The scatter plots show the correlation between Date Fragmentation Ratio (DFR) and temporal reasoning accuracy for each language.

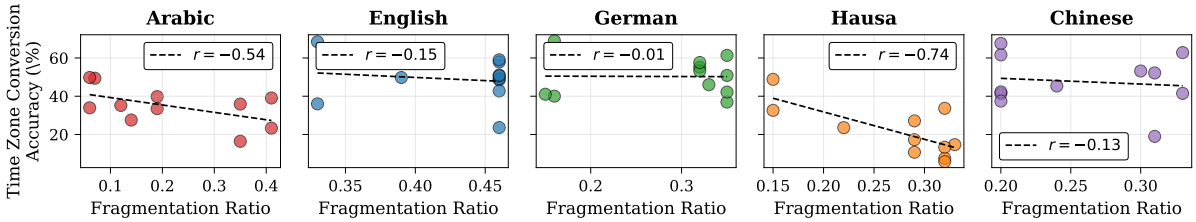
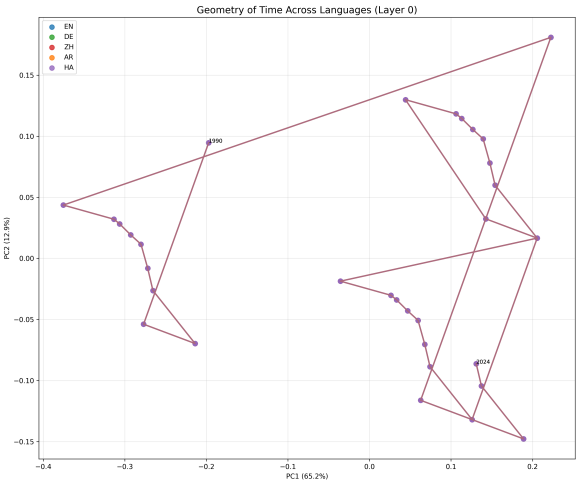


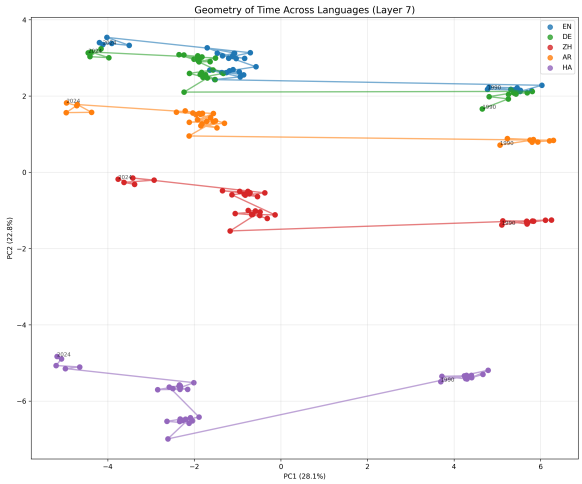
Figure 7: **Impact of Tokenisation on Time Zone Conversion Task Accuracy.** The scatter plots show the correlation between Date Fragmentation Ratio (DFR) and temporal reasoning accuracy for each language.

A.6 LLM as judge Prompts

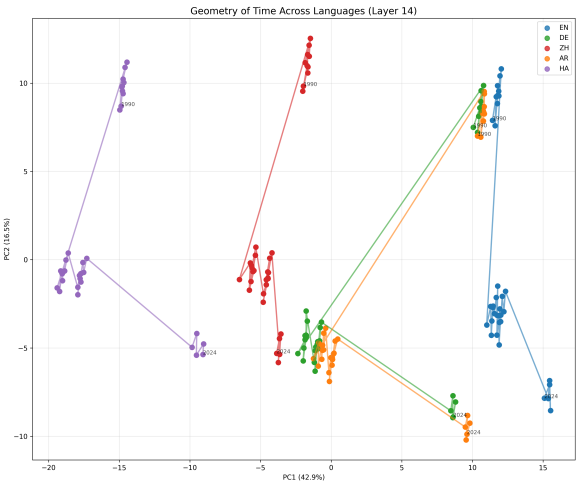
Our prompt for LLM-as-judge is illustrated in Table 10.



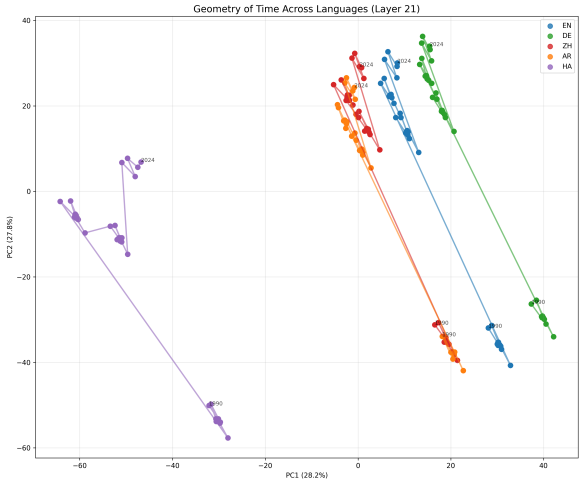
(a) **Layer 0 (Input):** Chaotic trajectories dominated by surface-level tokenisation fragmentation.



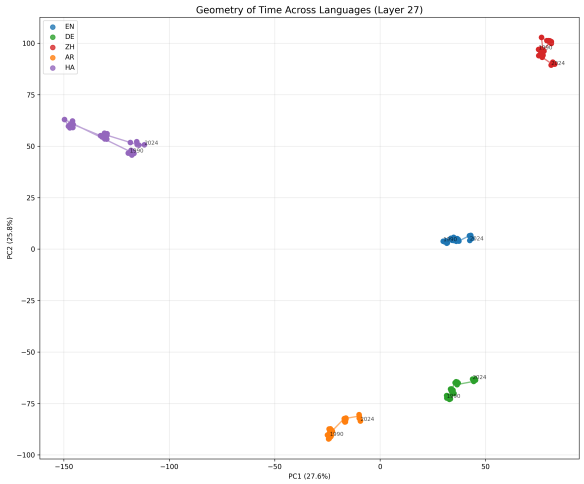
(b) **Layer 7 (Early):** Representations separate by language syntax; global linear organization has not yet formed.



(c) **Layer 14 (Middle):** High-resource languages (EN, DE, ZH) begin to straighten; low-resource (HA) remains curved.



(d) **Layer 21 (Reasoning):** The *Geometric Language Tax*. EN/DE/ZH form near-linear year trajectories useful for arithmetic, while HA remains a non-linear cluster.



(e) **Layer 27 (Output):** Final separation of language clusters to prepare for distinct lexical decoding.

Figure 8: **Evolution of temporal organization across layers.** PCA projections of year centroid embeddings (1990–2024) in Qwen 3. The plots show a progression from input-level fragmentation (Layer 0) to temporally structured, approximately linear trajectories in mid-to-deep layers for high-resource languages, while Hausa fails to linearize, remaining geometrically misaligned with the year-structured axis.

Human Evaluation

Context-based resolution

Prompt: Who was the chair of Allgemeiner Deutscher Fahrrad-Club in 17/10/2016?
Gold Answer: Ulrich Syberg
Model Prediction: As of October 17, 2016, the Federal Chairman was Ulrich Syberg

Human Annotator Rating: CORRECT

LLM-as-Judge Rating: CORRECT

Date arithmetic

Prompt: What date is 60 days after 05/01/1225?
Gold Answer: March 6, 1225 , June 29, 1225
Model Prediction: July 30, 1225

Human Annotator Rating: INCORRECT

LLM-as-Judge Rating: INCORRECT

Table 9: Human evaluation of LLM-as-judge.

LLM-as-Judge Evaluation Prompt

Your task: Evaluate one prediction at a time. You receive:

- **Question** – the task prompt shown to the model
- **Gold target** – *all* answers that are considered correct
- **Predicted answer** – the model's response

Return **one letter only**:

A	CORRECT	prediction fully matches *one* gold variant
B	INCORRECT	prediction contradicts or misses required info
C	NOT_ATTEMPTED	prediction refuses, guesses, or answers irrelevantly

General rules:

1. Match semantics, ignore capitalisation, punctuation, order.
2. If any statement contradicts the gold target, grade **B**.
3. Hedging ("I think. . . ") is fine if the correct info is present and no incorrect info is added.
4. Partial answers are **B**. Typos that preserve meaning are allowed.

DateAugBench specifics:

- **Date format ambiguity**: gold lists every valid interpretation; accept any.
- **Date arithmetic**: prediction must match *day, month, year* of a listed variant, any textual format allowed.
- **Format-switch questions**: answer with any synonym of `Yes/True` or `No/False`.
- **Numeric answers** – must match the gold number to the last shown significant digit.

Output format
Return exactly one capital letter:

`A` or `B` or `C`

No additional text or punctuation.

Example template

```
Question: {question}
Gold target: {target}
Predicted answer: {predicted_answer}
```

Now grade:

`A` or `B` or `C`

Table 10: LLM-as-Judge prompt used for comparing model and gold answers in the three tasks in MULTITEMPBENCH.