Chronocept: Instilling a Sense of Time in Machines

Krish Goel*, **Sanskar Pandey***, **KS Mahadevan, Harsh Kumar, Vishesh Khadaria**
krish@littlebird.ai, pandeysanskar854@gmail.com, mahadevanks26@gmail.com,
kumarharsh3014@gmail.com, khadariavishesh@gmail.com

Abstract

Human cognition is deeply intertwined with a sense of time, known as *Chronoception*. This sense allows us to judge how long facts remain valid and when knowledge becomes outdated. Despite progress in vision, language, and motor control, AI still struggles to reason about temporal validity. We introduce Chronocept, the first benchmark to model temporal validity as a continuous probability distribution over time. Using skew-normal curves fitted along semantically decomposed temporal axes, Chronocept captures nuanced patterns of emergence, decay, and peak relevance. It includes two datasets: Benchmark I (atomic facts) and Benchmark II (multi-sentence passages). Annotations show strong inter-annotator agreement (84% and 89%). Our baselines predict curve parameters - location, scale, and skewness - enabling interpretable, generalizable learning and outperforming classification-based approaches. Chronocept fills a foundational gap in AI's temporal reasoning, supporting applications in knowledge grounding, fact-checking, retrieval-augmented generation (RAG), and proactive agents. Code and data are publicly available.

1 Introduction

Humans effortlessly track how information changes in relevance over time. We instinctively know when facts emerge, become useful, or fade into obsolescence - a cognitive ability known as Chronoception (Fontes et al., 2016; Zhou et al., 2019). This higher-order perception of time plays a crucial role in how we evaluate the persistence and usefulness of information in real-world contexts. Despite excelling in pattern recognition (He et al., 2016), language generation (Brown et al., 2020), and motor control (Levine et al., 2016), modern AI systems remain largely insensitive to the temporal validity of the information they process.

Prior work has advanced temporal understanding via event ordering (Allen, 1983; Ning et al., 2020a; Wen and Ji, 2021), timestamp prediction (Kanhabua and Nørvåg, 2008; Kumar et al., 2012; Das et al., 2017), and temporal commonsense reasoning (Zhou et al., 2019). However, these approaches often reduce time to static labels or binary transitions. Even recent efforts in temporal validity change prediction (Wenzel and Jatowt, 2024) model shifts as discrete class changes, neglecting the gradual and asymmetric nature of temporal decay.

We introduce Chronocept, a benchmark that models temporal validity as a continuous probability distribution over time. Using a skewed-normal distribution over logarithmic time, parameterized by location (ξ), scale (ω), and skewness (α) (Azzalini, 1986; Schmidt et al., 2017), Chronocept captures subtle temporal patterns such as delayed peaks and asymmetric decay.

To support structured supervision, we decompose each sample along semantic temporal axes. We release two benchmarks: Benchmark I features atomic factual statements, and Benchmark II contains multi-sentence passages with temporally interdependent elements. High inter-annotator agreement across segmentation, axis labeling, and curve parameters validates annotation quality.

We benchmark modern encoder families - RoBERTa (Liu et al., 2019b), DeBERTa-v3 (He et al., 2021), and DistilBERT (Sanh et al., 2019) - alongside sentence-embedding heads (SBERT-FFNN, SBERT-BiLSTM) (Reimers and Gurevych, 2019) and a multi-task head (MT-DNN) (Liu et al., 2019a). The models are trained using a unified parameter-space Gaussian NLL objective over $\xi \in \mathbb{R}$, $\log \omega \in \mathbb{R}$, and $\tilde{\alpha} = \operatorname{artanh}(\alpha/A) \in \mathbb{R}$, which introduces learned heteroscedastic uncertainties for each parameter. This geometry-aware formulation stabilizes optimization and calibrates the relative sensitivity of scale and skewness without assuming

*Equal contribution.

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 4: Student Research Workshop, pages 437–456

any observational distribution.

To analyze key design factors, we conduct ablations on three fronts: (i) axis encoding, (ii) objective loss formulation, and (iii) axis shuffling to test structural sensitivity.

Chronocept enables several downstream applications. In Retrieval-Augmented Generation (RAG), temporal curves guide time-sensitive retrieval; in fact-checking, they help flag decaying or stale facts. Most importantly, Chronocept lays the foundation for proactive AI systems that reason not just about what to do, but when to do it (Miksik et al., 2020).

All resources - dataset, and baseline implementations - are publicly available to support future research in machine time-awareness.

2 Related Work

2.1 Temporal Validity Prediction

In the earliest attempt to formalize the temporal validity of information, Takemura and Tajima (2012) proposed the concept of "content viability" by classifying tweets into "read now," "read later," and "expired" categories, to prioritize timeliness in information consumption. However, their approach assumed a rigid, monotonic decay of relevance, failing to model scenarios where validity peaks later rather than at publication. This restricted its applicability beyond real-time contexts such as Twitter streams.

Almquist and Jatowt (2019) extended this work by defining a "validity period" and effectively proposing a "content expiry date" for sentences, using linguistic and statistical features. However, their reliance on static time classes (e.g., hours, days, weeks) sacrificed granularity, and their approach required explicit feature engineering rather than leveraging more advanced, data-driven methods (Das et al., 2017).

Traditional approaches (Almquist and Jatowt, 2019; Lynden et al., 2023; Hosokawa et al., 2023) mostly treat validity as binary, where information is either valid or invalid at a given time, this can be formulated as:

$$\text{validity}_i(t) = \begin{cases} \text{True} & \text{if information } i \text{ is valid at } t, \\ \text{False} & \text{otherwise} \end{cases} \quad (1)$$

where i represents the information under consideration and t denotes the time at which its validity is evaluated. However, this model overlooks gradual decay, recurrence, and asymmetric relevance patterns.

More recently, Wenzel and Jatowt (2024) introduced Temporal Validity Change Prediction (TVCP), which models how context alters a statement's validity window. However, it does not quantify validity as a continuous probability distribution over time.

Chronocept advances this field by defining temporal validity as a continuous probability distribution, allowing a more precise and flexible representation of how information relevance evolves.

2.2 Temporal Reasoning and Commonsense

Temporal reasoning has largely focused on event ordering (Allen, 1983; Wen and Ji, 2021; Ning et al., 2020a), predicting temporal context (Kanhabua and Nørvåg, 2008; Kumar et al., 2012; Das et al., 2017; Luu et al., 2021; Jatowt et al., 2013), and commonsense knowledge (Zhou et al., 2019). While these studies laid the groundwork for understanding event sequences, durations, and frequencies, recent work has expanded into implicit or commonsense dimensions of temporal reasoning.

TORQUE (Ning et al., 2020b) is a benchmark designed for answering temporal ordering questions, while TRACIE, along with its associated model SYMTIME (Zhou et al., 2021), primarily ensures temporal-logical consistency rather than modeling truth probabilities.

MCTACO (Zhou et al., 2019) evaluates temporal commonsense across five dimensions: event duration, ordering, frequency, stationarity, and typical time of occurrence. MCTACO assesses whether a given statement aligns with general commonsense expectations, and does not quantify the likelihood of a statement being true over time.

Recent work (Wenzel and Jatowt, 2023; Jain et al., 2023) has explored how LLMs handle temporal commonsense, exposing inconsistencies in event sequencing and continuity. However, these studies do not incorporate probabilistic modeling of temporal validity - a core focus of Chronocept, which models truthfulness as a dynamic, evolving probability distribution.

2.3 Dataset Structuring for Temporal Benchmarks

Temporal annotation frameworks like TIMEML (Pustejovsky et al., 2003) and ISO-TIMEML (Pustejovsky et al., 2010) focus on static event relationships, often suffering from low inter-annotator agreement due to event duration ambiguities. The TIMEBANK series (Pustejovsky, 2003; Cassidy

et al., 2014) and TEMPEVAL challenges (Verhagen, 2007, 2010; UzZaman et al., 2012) expanded evaluations but remained limited in modeling evolving event validity.

In response, Ning et al. (2018) proposed a multi-axis annotation scheme (MATRES) that structures events into eight distinct categories - Main, Intention, Opinion, Hypothetical, Negation, Generic, Static, and Recurrent. Additionally, the scheme prioritizes event start points over full event intervals, reducing ambiguity and significantly improving IAA scores. Chronocept builds on this by refining multi-axis annotation to model temporal validity, capturing how information relevance shifts over time through probabilistic distributions.

2.4 Statistical Modeling of Temporal Data Using Skewed Normal Distribution

Traditional normal distributions often fail to capture skewed temporal patterns. The skew-normal distribution (Azzalini, 1986, 1996) provides a more flexible alternative by incorporating asymmetry, improving accuracy in modeling time-dependent information relevance (Schmidt et al., 2017). Chronocept employs this distribution to capture various temporal behaviors, including gradual decay, peak relevance, and rapid obsolescence.

3 Chronocept: Task & Benchmark Design

3.1 Problem Definition

Temporal Validity Prediction (TVP) models how the validity of information evolves over time after publication. Unlike prior work that formulates TVP as a discrete classification problem, we model temporal validity as a continuous-time marginal probability function.

Let $T \subseteq \mathbb{R}_{\geq 0}$ denote elapsed time since publication of information i. We define a binary random variable

$$\text{validity}_i(t) \in \{0, 1\} \tag{2}$$

where $\text{validity}_i(t) = 1$ indicates that i is valid at time t. The instantaneous marginal probability of validity is

$$p_i(t) \triangleq P(\text{validity}_i(t) = 1),\ p_i : T \to [0, 1] \tag{3}$$

For any interval $[a, b]$, the integral $\int_a^b p_i(t)\, dt$ corresponds to the expected total duration for which the statement is valid within that interval, or equivalently, the probability that a uniformly sampled time point from $[a, b]$ is valid (up to normalization). Importantly, this quantity does not represent the joint probability of uninterrupted validity over the entire interval.

We impose no boundary conditions such as $p_i(0) = 1$ or monotonic decay, allowing the model to capture delayed onset, non-monotonic decay, and other complex temporal validity patterns (Takemura and Tajima, 2012; Almquist and Jatowt, 2019).

3.2 Modeling Temporal Validity

We model the temporal validity of statements using a probability curve, with likelihood of being valid on the Y-axis and time since publication on the X-axis. To reduce ambiguity, sentences are decomposed along semantically distinct axes. A skew-normal distribution on a logarithmic time scale captures the validity dynamics.

Axes-Based Decomposition. We use the multi-axis annotation framework of Ning et al. (2018) (MATRES), which partitions each sentence into eight semantically distinct axes - Main, Intention, Opinion, Hypothetical, Generic, Negation, Static, and Recurrent.

This decomposition confines relation annotation within axis-specific contexts, reducing cross-category interference and improving inter-annotator agreement. It provides cleaner supervision by isolating coherent temporal semantics and guiding models toward human-aligned representations of temporal structure. In our ablation (Appendix F), removing axis features increases MSE by 5.95%, indicating that axis-level signals also contribute to temporal precision.

Skewed Normal Distribution. We model temporal validity using the skewed normal distribution, a generalization of the Gaussian with a shape parameter α that captures asymmetry. This enables representation of non-symmetric temporal patterns such as delayed onset, gradual decay, or skewed relevance, which symmetric (Gaussian) or memoryless (exponential) distributions fail to model.

The probability density function is:

$$f(x; \xi, \omega, \alpha) = \frac{2}{\omega}\, \phi\left(\frac{x - \xi}{\omega}\right)\, \Phi\left(\alpha\, \frac{x - \xi}{\omega}\right) \tag{4}$$

where:

- $\phi(z)$ is the standard normal PDF,

- $\Phi(z)$ is the standard normal CDF,
- ξ is the location parameter - determining the time at which an event is most likely valid,
- ω is the scale parameter - governing the duration of validity, and
- α is the shape parameter - controlling skewness (with positive values yielding right skew and negative values left skew).

Quantitative comparisons against Gaussian, log-normal, exponential and gamma distributions in Appendix D support this choice.

Logarithmic Time Scale. Linear time yields sparse coverage over key intervals, particularly at minute-level granularity. To address this, we compress the time axis using a monotonic logarithmic transformation:

$$t' = \log_{1.1}(t) \tag{5}$$

We default to a base of 1.1 for the near-linear spacing across canonical intervals (e.g., minutes, days, decades) while preserving granularity. Chronocept's target values remain compatible with alternative bases. See Appendix C for the base transformation framework, compression analysis, and the provided code implementation.

4 Dataset Creation

4.1 Benchmark Generation & Pre-Filtering

Chronocept comprises two benchmarks to facilitate evaluation across varying complexity levels. Benchmark I consists of 1,254 samples featuring simple, single-sentence texts with clear temporal relations - ideal for baseline reasoning - while Benchmark II includes 524 samples with complex, multi-sentence texts capturing nuanced, interdependent temporal phenomena.

Synthetic samples were generated using the GPT-o1[1] model (OpenAI, 2024) with tailored prompts to ensure temporal diversity across benchmarks. Full prompts for both benchmarks are disclosed in Appendix E for reproducibility. No real-world or personally-identifying data was used, ensuring complete privacy.

In the pre-annotation phase, SBERT[2] (Reimers and Gurevych, 2019) and TF-IDF embeddings were generated for all samples, and pairwise cosine similarities were calculated. Samples with SBERT or TF-IDF similarity exceeding 0.7 (70%) were removed to reduce semantic and lexical redundancy.

Annotation guidelines are disclosed in Appendix A and were continuously accessible during annotation.

4.2 Annotation Workflow

Annotation Process. Our protocol consists of three steps: (i) *Temporal Segmentation* – partitioning text into coherent subtexts that preserve temporal markers; (ii) *Axis Categorization* – assigning each segment to one of eight temporal axes (Main, Intention, Opinion, Hypothetical, Generic, Negation, Static, Recurrent); and (iii) *Temporal Validity Distribution Plotting* – annotating a skewed normal distribution, parameterized by location (ξ), scale (ω), and skewness (α), over a logarithmic time axis.

To ensure interpretability and consistency, all parent texts are written in the present tense, distributions are anchored at $t = 0$, and multimodal curves are excluded. Additionally, any samples that did not exhibit a clearly assignable main timeline or violated these constraints were flagged and discarded during the annotation process.

4.3 Annotator Training & Quality Control

Eight third-year B.Tech. students with relevant coursework in Natural Language Processing, Machine Learning, and Information Retrieval participated. They underwent a 1-hour training session and a supervised warm-up on 50 samples. Agreement thresholds were set at ICC > 0.90 for numerical annotations, Jaccard Index > 0.75 for segment-level annotations, and $P_k < 0.15$ for segmentation consistency during this warm-up phase.

Each sample was annotated independently by two annotators. Quality control included daily reviews of 10% of annotations, a limit of 70 samples per annotator per day to mitigate fatigue, and automated flagging of samples with segmentation mismatches, target deviations >2σ, or $P_k > 0.2$. Discrepancies were adjudicated or, if unresolved, discarded.

No personal or identifying information was collected or stored during the annotation process.

Handling Edge Cases and Final Resolution. Ambiguous samples were flagged or discarded following the three-phase filtering scheme. For segmentation and axis labels, a union-based

[1] https://openai.com/o1

[2] all-MiniLM-L6-v2 available at https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

approach retained all plausible interpretations, recognizing that axis confusion may encode aspects of human temporal cognition useful for future modeling. For temporal validity targets (ξ, ω, α), annotator values were averaged to yield smooth probabilistic supervision rather than discrete target selection.

4.4 Inter-Annotator Agreement (IAA)

We evaluate Inter-Annotator Agreement (IAA) using stage-specific metrics aligned with each step of the annotation task. Segmentation quality is assessed using the P_k metric (Beeferman et al., 1997), axis categorization consistency is measured using the Jaccard Index, and agreement on the final temporal validity parameters (ξ, ω, α) is quantified using the Intraclass Correlation Coefficient (ICC).

We report only ICC as the benchmark-wide IAA, refraining from aggregating agreement across stages, as segmentation and axis categorization, while enriching the dataset structure, do not directly impact the core prediction task, which depends solely on the parent text and its annotated temporal validity distribution.

Agreement statistics across both benchmarks are summarized in Table 1. We observed notable confusion between the *Generic* and *Static* axes during the early stages of annotation, particularly in the warm-up phase. This source of disagreement is analyzed in detail in Appendix B.

IAA Metric	BI	BII
ICC	0.843	0.893
Jaccard Index	0.624	0.731
P_k **Metric**	0.233	0.009

Table 1: IAA metrics for segmentation, axis categorization, and temporal validity annotation across both benchmarks. For P_k, lower is better, with values ranging from 0 (perfect agreement) to 1 (chance-level).

4.5 Dataset Design

Each Chronocept sample captures the temporal dynamics of factual information through a structured annotation format, as illustrated in Figure 1.

Parent Text. A single sentence serving as the basis for annotation.

Temporal Axes. Each parent text is segmented into subtexts annotated along eight temporal axes:

- **Main:** Core verifiable events.
- **Intention:** Future plans or desires.
- **Opinion:** Subjective viewpoints.
- **Hypothetical:** Conditional or imagined events.
- **Negation:** Denied or unfulfilled events.
- **Generic:** Timeless truths or habitual patterns.
- **Static:** Unchanging states in context.
- **Recurrent:** Repeated temporal patterns.

Target Values. Temporal validity is quantified by three parameters:

- ξ **(Location):** The time point of peak validity.
- ω (Scale): The duration over which validity is maintained.
- α **(Skewness):** The asymmetry of the validity curve.

4.6 Dataset Statistics & Splits

Stratified sampling over the axes distribution was applied to partition the datasets into training (70%), validation (20%), and test (10%) splits, ensuring equitable axis coverage. Table 2 summarizes the splits for both benchmarks. The axes distribution, calculated based on non-null annotations for each sample, is detailed in Table 3.

Benchmark	Training	Validation	Test
Benchmark I	878	247	129
Benchmark II	365	104	55

Table 2: Dataset Composition and Splits.

Token-level[3] and target parameter-level statistics for both benchmarks are summarized in Table 4 and Table 5.

4.7 Accessibility and Licensing

The Chronocept dataset is released under the Creative Commons Attribution 4.0 International (CC-BY 4.0)[4] license, allowing unrestricted use with proper attribution. It is publicly available on Hugging Face Datasets at: https://huggingface.co/datasets/krishgoel/chronocept.

[3]Tokenization performed using SpaCy's `en_core_web_sm` model: https://spacy.io/api/tokenizer

[4]https://creativecommons.org/licenses/by/4.0

```
{
    "_id": "H0028",
    "parent_text": "They are discussing a philosophical concept, whereas an online forum
simultaneously erupts in debate over similar ideas. They believe open dialogue fosters
clarity, yet they recognize tensions may escalate. They intend to document their
conclusions, hoping to contribute thoughtfully to the discussion."
    "axes": {
        "main_outcome_axis": "They are discussing a philosophical concept,",
        "intention_axis": "They intend to document their conclusions, hoping to
contribute thoughtfully to the discussion.",
        "opinion_axis": "They believe open dialogue fosters clarity,",
        "hypothesis_axis": "",
        "generic_axis": "",
        "negation_axis": "",
        "static_axis": "whereas an online forum simultaneously erupts in debate over
similar ideas. yet they recognize tensions may escalate.",
        "recurrent_axis": ""
    },
    "target_values": {
        "location": 39.865,
        "scale": 13.265,
        "skewness": 4.25
    }
}
```


Figure 1: Composition of samples in Chronocept benchmarks.

Temporal Axis	Benchmark I	Benchmark II
Main Axis	1254	524
Static Axis	516	513
Generic Axis	228	116
Hypothetical Axis	136	182
Negation Axis	240	200
Intention Axis	165	522
Opinion Axis	328	519
Recurrent Axis	348	198

Table 3: Distribution of annotated temporal axes across Benchmark I and Benchmark II.

Benchmark	Mean Length (μ)	SD (σ)
Benchmark I	16.41 tokens	1.56 tokens
Benchmark II	56.21 tokens	6.21 tokens

Table 4: Sentence Length Statistics for Benchmarks.

5 Baseline Model Performance

5.1 Task Scope and Evaluation Focus

Chronocept frames temporal validity as structured regression over low-dimensional parameters - location (ξ), scale (ω), and skewness (α) - predicted from annotated parent texts. In contrast to event ordering (Pustejovsky, 2003), commonsense classification (Zhou et al., 2019), or temporal shift detection (Wenzel and Jatowt, 2024), segmentation and axis labels are preprocessing artifacts and are not inferred at test time.

Evaluation reports MSE, MAE, RMSE, and R^2 for accuracy; NLL for calibration; and Spearman correlation (ρ) for rank agreement.

Evaluation units & normalization. NLL values are not directly comparable across objective families: parameter-space models compute Gaussian NLL on z-scored parameters $(\xi, \log\omega, \tilde{\alpha})$, while label-space outputs are unnormalized and MSE-trained models lack explicit variance terms, inflating post-hoc NLL. We therefore compare NLL only within objective families; use MSE/CRPS for cross-objective comparisons.

5.2 Baseline Models and Training Setup

We evaluate six encoder architectures: RoBERTa (Liu et al., 2019b), DeBERTa-v3 (He et al., 2021), DistilBERT (Sanh et al., 2019); a multi-task MT-DNN (Liu et al., 2019a); and SBERT with two heads - SBERT-FFNN and SBERT-BiLSTM (Reimers and Gurevych, 2019). All mod-

Parameter	Location (ξ)		Duration (ω)		Skewness (α)	
Benchmark	**Mean (μ)**	**SD (σ)**	**Mean (μ)**	**SD (σ)**	**Mean (μ)**	**SD (σ)**
Benchmark I	54.2803	20.4169	11.5474	3.7725	-0.0158	1.3858
Benchmark II	46.1511	13.3839	9.5553	2.5725	0.0275	1.1773

Table 5: Temporal Parameter Distribution Statistics for Benchmarks.

els jointly predict (ξ, ω, α) with a parameter-space Gaussian negative log-likelihood (Gaussian NLL); ablations compare label-space `MSE`, `Gaussian NLL`, and `Skew-Normal NLL`. Training uses 100 epochs, a five-epoch warm start on ξ, AdamW, and layer-wise learning-rate decay $\text{lr}_{\text{head}} = 5 \times \text{lr}_{\text{encoder}}$. Targets are z-score normalized per benchmark.

All experiments were conducted on a machine with an Intel Core i9-14900K CPU, 16GB DDR5 RAM, and an NVIDIA RTX 4060 GPU.[5]

5.3 Quantitative Evaluation

Table 6 reports aggregate results on BI and BII under a unified configuration (Gaussian NLL, 100 epochs, five-epoch warm start). MT-DNN attains the best pointwise accuracy across both benchmarks (lowest MSE/MAE; strongest R^2), while SBERT-FFNN yields the lowest NLL on BI and BII. DeBERTa-v3, DistilBERT, and RoBERTa trail on aggregate error but show parameter-specific strengths. Negative R^2 values indicate performance below a mean predictor under noise and limited data and should be interpreted comparatively across baselines.

Per-parameter analysis (Table 7) refines these trends. MT-DNN achieves the lowest RMSE for ξ on both benchmarks; for ω, DistilBERT is lowest on BI and RoBERTa on BII; for α, DeBERTa-v3 leads on BI and RoBERTa on BII. Rank correlations are heterogeneous: MT-DNN has the highest ρ for ξ on BI, SBERT-FFNN leads for ξ on BII; DeBERTa-v3 is strongest for ω on BII; DistilBERT leads for α on BII. Overall, the parameter-space objective gives robust point estimates (especially for ξ), while sentence-embedding heads can offer competitive calibration.

5.4 Ablations: Temporal Axes and Objective Choice

Structured temporal axes. Encoding axis structure improves fit and calibration across backbones. With RoBERTa fixed, representation-level fusion via SBERT concatenation yields the largest gains: on BI, NLL decreases by 90.66% and MSE by 5.95%; on BII, NLL decreases by 65.30%. Inline single-sequence markers provide smaller BI gains and mixed calibration on BII (see Table 12, Table 13; extended analysis in Appendix F).

Axis-order robustness. Maintaining axis order is critical. Training with shuffled axis order (evaluation unperturbed) worsens metrics relative to the ordered control: MSE +0.87%, MAE +1.09%, R^2 6.31% more negative, and NLL +358.15% with RoBERTa (see Table 14; full study in Appendix G).

Objective function. We compare label-space losses (MSE, Gaussian NLL, Skew-Normal NLL) against a geometry-aware parameter-space Gaussian NLL. While label-space MSE minimizes error, it yields severe miscalibration (BI NLL ≈220; BII ≈271); the parameter-space objective over $\boldsymbol{\theta} = (\xi, \log\omega, \operatorname{artanh}(\alpha/A))$ provides the most stable NLL with strong accuracy (see Table 15, Table 16, Table 17, Table 18 and Appendix H). Skewness is bounded via $\alpha = A\tanh(\tilde{\alpha})$ with $A = 5$. Gaussian NLL assumes Gaussian residuals only in normalized parameter space, not a generative model of temporal validity.

6 Conclusion & Applications

We introduced Chronocept, a framework that models temporal validity as a continuous probability distribution using a unified, parameterized representation. By encoding validity through location (ξ), scale (ω), and skewness (α), it offers a generalizable basis for temporal reasoning in language.

Structured annotations and explicit temporal axes enable models to capture not only *if*, but also *when* and *for how long* information remains valid - moving beyond binary truth labels toward richer temporal understanding.

Empirical results show that simple neural encoders paired with pretrained embeddings (MT-

[5]Code and scripts: https://github.com/krishgoel/chronocept-baseline-models.

Metric	MSE		MAE		R^2		NLL	
Baseline	BI	BII	BI	BII	BI	BII	BI	BII
DeBERTa-v3	695.7505	544.5860	14.8318	13.7358	-1.3404	-2.7730	9.9973	11.6107
Z DistilBERT	737.3157	640.2664	15.3509	14.7484	-1.5092	-3.0546	11.3089	13.6647
MT-DNN	**108.3768**	**64.5708**	**5.9425**	**4.5253**	**0.0314**	**-0.0183**	4.5117	4.1865
RoBERTa	909.5181	748.8589	17.2330	16.0097	-1.8687	-3.5666	15.2214	15.2253
SBERT-BiLSTM	171.7044	107.4771	8.5698	6.0472	-2.3193	-1.1621	4.4084	4.1373
SBERT-FFNN	155.8747	153.2178	7.9780	8.5744	-2.0203	-6.3268	**4.3769**	**3.9842**

Table 6: Performance of encoder-based baselines on Benchmark I (BI) and Benchmark II (BII). Lower MSE, MAE, and NLL indicate better fit; higher R^2 denotes stronger alignment between predicted and true temporal parameters.

Metric		RMSE			Spearman		
Baseline	**Benchmark**	ξ	ω	α	ξ	ω	α
DeBERTa-v3	Benchmark I	45.5250	3.6120	**1.2950**	-0.1179	0.0318	**0.1953**
	Benchmark II	40.2941	2.9834	1.1133	-0.2608	**0.3994**	0.0037
DistilBERT	Benchmark I	46.8732	**3.5750**	1.4375	-0.0545	0.0763	0.0860
	Benchmark II	43.7430	2.5137	1.0161	0.0754	0.2373	**0.2271**
MT-DNN	Benchmark I	**17.6030**	3.6852	1.2974	**0.5200**	0.1405	0.0115
	Benchmark II	**13.6704**	2.4017	1.0319	0.1807	-0.0488	0.0042
RoBERTa	Benchmark I	52.0884	3.6933	1.3079	-0.1243	-0.0371	0.0948
	Benchmark II	47.3300	**2.3293**	**1.0111**	0.0882	0.2670	0.1383
SBERT-BiLSTM	Benchmark I	20.7577	8.9916	1.8386	0.3710	0.1396	-0.0385
	Benchmark II	17.3793	4.3467	1.2238	0.1337	-0.2033	-0.0327
SBERT-FFNN	Benchmark I	19.6840	8.8119	1.5859	0.4687	**0.1670**	-0.0295
	Benchmark II	19.5139	8.4704	2.6672	**0.3690**	-0.0511	-0.1867

Table 7: Per-parameter RMSE (accuracy) and Spearman ρ (monotonic agreement) for all baselines.

DNN) perform effectively, while ablations highlight the importance of structural consistency and axis-level decomposition.

Chronocept opens avenues for temporally aware applications such as retrieval-augmented generation, fact verification, knowledge lifecycle modeling, and proactive AI agents that act on temporal salience (Miksik et al., 2020). All datasets, annotations, and baselines are publicly released to support future research.

Limitations

Unimodal Representation. Chronocept models temporal validity as a single-peaked distribution - interpretable but unable to capture phenomena with multiple or recurring relevance periods.

Sentence-Level Scope. The dataset comprises short, self-contained sentences without document-level or historical context, limiting the modeling of long-range temporal dependencies.

Lack of Atemporality Labels. The absence of explicit markers for universally valid or atemporal facts creates ambiguity between permanent and time-sensitive information.

Minimum Validity Bound. The logarithmic time scale imposes a lower limit of one minute, making Chronocept unsuitable for instantly obsolete events such as flash updates or ephemeral statements.

7 Acknowledgments

We thank Mohammed Iqbal, Meenakshi Kumar, Yudhajit Mondal, Tanish Sharma, Devansh Sharma, Lakshya Paliwal, Ishaan Verma, and Sanjit Chitturi for their help with data annotation.

References

James F Allen. 1983. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843.

Axel Almquist and Adam Jatowt. 2019. Towards content expiry date determination: Predicting validity periods of sentences. pages 86–101.

A Azzalini. 1996. The multivariate skew-normal distribution. *Biometrika*, 83(4):715–726.

Adelchi Azzalini. 1986. A class of distributions which includes the normal ones. *Scandinavian Journal of Statistics*.

Doug Beeferman, Adam Berger, and John Lafferty. 1997. Text segmentation using exponential models. In *Second Conference on Empirical Methods in Natural Language Processing*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Gaurav Sastry, Amanda Askell, Ariel Agarwal, Shelly Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. 33:1877–1901.

Taylor Cassidy, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering.

Supratim Das, Arunav Mishra, Klaus Berberich, and Vinay Setty. 2017. Estimating event focus time using neural word embeddings. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, New York, NY, USA. ACM.

Rhailana Fontes, Jéssica Ribeiro, Daya S Gupta, Dionis Machado, Fernando Lopes-Júnior, Francisco Magalhães, Victor Hugo Bastos, Kaline Rocha, Victor Marinho, Gildário Lima, Bruna Velasques, Pedro Ribeiro, Marco Orsini, Bruno Pessoa, Marco Antonio Araujo Leite, and Silmar Teixeira. 2016. Time perception mechanisms at central nervous system. *Neurol. Int.*, 8(1):5939.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. pages 770–778.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Taishi Hosokawa, Adam Jatowt, and Kazunari Sugiyama. 2023. Temporal natural language inference: Evidence-based evaluation of temporal text validity. In *Lecture Notes in Computer Science*, Lecture notes in computer science, pages 441–458. Springer Nature Switzerland, Cham.

Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do language models have a common sense regarding time? revisiting temporal commonsense reasoning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6750–6774, Stroudsburg, PA, USA. Association for Computational Linguistics.

Adam Jatowt, Ching-Man Au Yeung, and Katsumi Tanaka. 2013. Estimating document focus time. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13*, New York, New York, USA. ACM Press.

Nattiya Kanhabua and Kjetil Nørvåg. 2008. Improving temporal language models for determining time of non-timestamped documents. In *Research and Advanced Technology for Digital Libraries*, Lecture notes in computer science, pages 358–370. Springer Berlin Heidelberg, Berlin, Heidelberg.

Abhimanu Kumar, Jason Baldridge, Matthew Lease, and Joydeep Ghosh. 2012. Dating texts without explicit temporal cues. *arXiv [cs.CL]*.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. 2016. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A Smith. 2021. Time waits for no one! analysis and challenges of temporal misalignment. *arXiv [cs.CL]*.

Steven Lynden, Mehari Heilemariam, Kyoung-Sook Kim, Adam Jatowt, Akiyoshi Matono, Hai-Tao Yu, Xin Liu, and Yijun Duan. 2023. Commonsense temporal action knowledge (cotak) dataset. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM 2023*.

Ondrej Miksik, I Munasinghe, J Asensio-Cubero, S Reddy Bethi, ST Huang, S Zylfo, Xuechen Liu, T Nica, A Mitrocsak, S Mezza, et al. 2020. Building proactive voice assistants: When and how (not) to interact. *arXiv preprint arXiv:2005.01322*.

Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020a. TORQUE: A reading

comprehension dataset of temporal ordering questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Stroudsburg, PA, USA. Association for Computational Linguistics.

Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020b. TORQUE: A reading comprehension dataset of temporal ordering questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1158–1172, Online. Association for Computational Linguistics.

Qiang Ning, Hao Wu, and Dan Roth. 2018. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1318–1328. Association for Computational Linguistics.

OpenAI. 2024. Openai o1 system card. *arXiv*.

J Pustejovsky, Kiyong Lee, H Bunt, and Laurent Romary. 2010. ISO-TimeML: An international standard for semantic annotation. *LREC*, pages 394–397.

James Pustejovsky. 2003. The timebank corpus. *Corpus linguistics*.

James Pustejovsky, José M Castaño, Robert Ingria, and Graham Katz. 2003. TimeML: A specification language for temporal and event expressions.

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)*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Alexandra M Schmidt, Kelly C M Gonçalves, and Patrícia L Velozo. 2017. Spatiotemporal models for skewed processes. *Environmetrics*, 28(6):e2411.

Hikaru Takemura and Keishi Tajima. 2012. Tweet classification based on their lifetime duration.

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. TempEval-3: Evaluating events, time expressions, and temporal relations. *arXiv [cs.CL]*.

Marc Verhagen. 2007. Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the fourth international workshop on semantic evaluations*.

Marc Verhagen. 2010. SemEval-2010 task 13: TempEval-2. In *Proceedings of the 5th international workshop on semantic evaluation*.

Haoyang Wen and Heng Ji. 2021. Utilizing relative event time to enhance event-event temporal relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA. Association for Computational Linguistics.

Georg Wenzel and Adam Jatowt. 2023. An overview of temporal commonsense reasoning and acquisition. *arXiv [cs.AI]*.

Georg Wenzel and Adam Jatowt. 2024. Temporal validity change prediction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1424–1446, Bangkok, Thailand. Association for Computational Linguistics.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. 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 3363–3369, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ben Zhou, Kyle Richardson, Qiang Ning, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2021. Temporal reasoning on implicit events from distant supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1361–1371, Online. Association for Computational Linguistics.

Appendix

A Annotation Guidelines

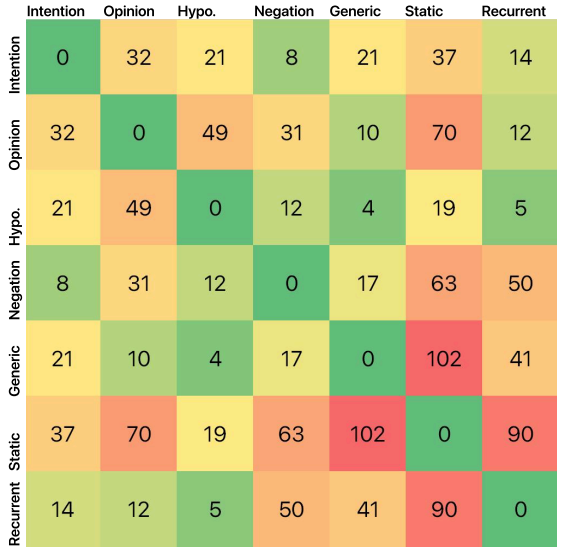
This section outlines the annotation guidelines used in the Chronocept dataset. These were introduced through an in-person training session and remained accessible throughout the annotation phase via a custom Streamlit-based interface for annotations[6]. The guidelines provide precise instructions for temporal segmentation, axis categorization, and temporal validity distribution plotting, supplemented with definitions, examples, and coverage of edge cases for all eight temporal axes.

During the initial warm-up phase, annotators exhibited substantial confusion between the Generic and Static axes. To mitigate this, the guidelines were revised to incorporate clearer contextual definitions and axis-specific "key questions" designed to improve disambiguation. These revisions led to a marked improvement in inter annotator agreement.

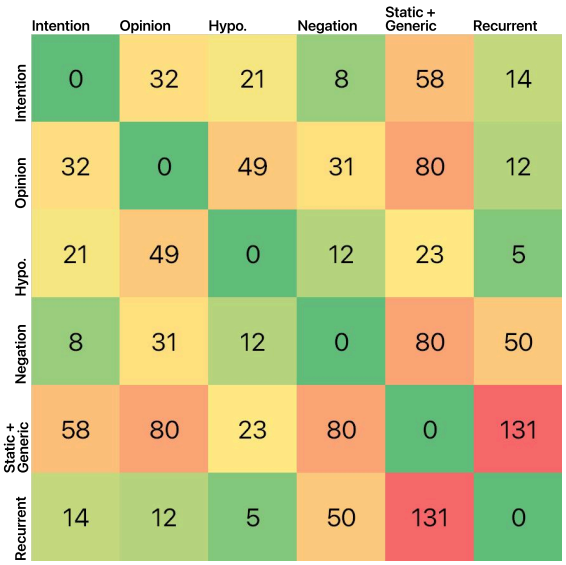
The complete guidelines are shown in Figure 2.

[6] https://streamlit.io

B Axis Confusion Analysis: Generic and Static



(a) Axis assignment co-occurrence matrix with Generic and Static treated as distinct classes



(b) Axis assignment co-occurrence matrix after merging Generic and Static into a unified class

Figure 3: Comparison of co-occurrence matrices before and after merging the Generic and Static axes, used to assess annotation consistency.

This appendix investigates a key source of annotator disagreement in the Chronocept annotation process: the difficulty in consistently distinguishing between the Generic and Static temporal axes.

Generic segments typically express habitual or timeless statements, while Static segments describe ongoing but context-specific states. Their semantic similarity led to frequent disagreement in axis assignment.

To address this, the annotation guidelines were refined during the warm-up phase with axis-specific clarifications and diagnostic questions. The guideline clarification led to reduced confusion, as shown in the co-occurrence matrices in Figure 3.

While co-occurrence matrices are traditionally used to analyze disagreement patterns between annotators, we treat them here as confusion matrices by including agreement counts along the diagonal, enabling standard metric computation.

To quantify the benefit of merging these axes, we computed micro-averaged inter-annotator precision. Treating this as a multi-class classification task, we additionally calculate Cohen's Kappa to assess inter-annotator agreement beyond chance. As shown in Table 8, merging resulted in a consistent improvement across both metrics: precision improved by 18.0% and Cohen's Kappa by 17.47%.

Axis Setting	Precision	Cohen's Kappa
Original	0.4443	0.3291
Merged	0.5243	0.3866

Table 8: Improvement in annotator alignment metrics after merging Generic and Static into a single class.

C Time Scale Logarithm Base Conversion

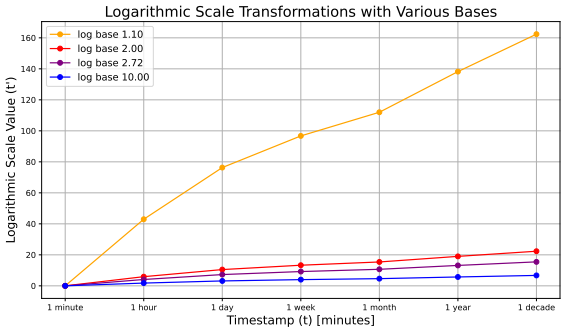


Figure 4: Effect of logarithmic base choice on time axis representation. Base 1.1 preserves quasi-linear spacing; larger bases induce stronger compression.

Chronocept represents time on a logarithmic axis to unify short- and long-term temporal dynamics in a compact space. The transformation is defined over a configurable base b; all released datasets use base 1.1. A reusable `DataLoader` with log conversion

is available in the official baselines repository[7].

Log Transformation. Given time t in minutes, the log-space representation is:

$$t' = \frac{\ln(t)}{\ln(b)}.$$

Base 1.1 yields quasi-linear spacing across intervals like hours, days, and years, preserving interpretability. Figure 4 shows that higher bases increasingly compress longer intervals, while base 1.1 maintains resolution across scales.

Compression Analysis. Table 9 summarizes the compression effect across bases 1.1, 2, and 10. For each timestamp, we report the log value t', compression ratio CR $= t'/t$, and percentage compression.

To convert values between log bases m and b:

$$t'^{(b)} = \frac{\ln(m)}{\ln(b)} \cdot t'^{(m)}.$$

Skew-Normal Parameter Adjustment. Chronocept models temporal validity using a skew-normal distribution:

$$f(x;\, \xi, \omega, \alpha) = \frac{2}{\omega}\, \phi\left(\frac{x-\xi}{\omega}\right)\, \Phi\left(\alpha\, \frac{x-\xi}{\omega}\right),$$

where ξ and ω denote location and scale. When converting between bases:

$$\xi^{(b)} = \frac{\ln(m)}{\ln(b)} \cdot \xi^{(m)}, \quad \omega^{(b)} = \frac{\ln(m)}{\ln(b)} \cdot \omega^{(m)}.$$

Skewness α remains invariant.

D Comparison of Distributions for Modeling Temporal Validity and Curve Fitting Methodology

This section evaluates candidate distributions for modeling temporal validity and outlines the curve fitting methodology. We consider six synthetic, unimodal scenarios varying along three axes: *offset* (peak position), *duration* (span of validity), and *asymmetry* (skew in rise and decay). Table 10 lists a representative sentence and five annotation points per scenario, placed on a base-1.1 logarithmic time axis.

Each temporal profile is defined by a smooth freehand curve from which five points are sampled - one at the peak, two mid-validity, and two low-validity points. These define a proportional shape used for fitting.

Unless otherwise stated, the temporal validity function $p_i(t)$ used in modeling and evaluation corresponds to the AUC-normalized fitted curve produced after optimization. Proportional or rescaled variants are derived only for numerical stability or visualization and are not used as probability functions.

Since these curves represent relative likelihoods provided by annotators, their area under the curve (AUC) is initially unconstrained. During optimization, a scaling factor is applied to fit freely, followed by Trapezoidal Rule normalization to enforce AUC = 1 while preserving shape.

To reduce computational overhead over long-tailed domains, we rescale the AUC-normalized fitted curve by its maximum value to constrain it to $[0, 1]$. This step is used solely for numerical stability or visualization and does not alter the underlying probabilistic semantics. All modeling, training, and evaluation rely on the AUC-normalized curve, which constitutes the temporal validity function $p_i(t)$.

Candidate distributions include:

Gaussian Normal:

$$f_{Gaussian}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Exponential:

$$f_{Exp}(x; \lambda) = \lambda \exp(-\lambda x), \text{where } x \geq 0$$

Log-normal:

$$f_{LN}(x; \mu, \sigma) = \frac{1}{x\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right),$$

$$\text{where } x > 0$$

Gamma:

$$f_{\Gamma}(x; k, \theta) = \frac{1}{\Gamma(k)\,\theta^k} x^{k-1} \exp\left(-\frac{x}{\theta}\right),$$

$$\text{where } x > 0$$

Skewed Normal:

$$f_{SN}(x; \xi, \omega, \alpha) = \frac{2}{\omega}\, \phi\left(\frac{x-\xi}{\omega}\right)\, \Phi\left(\alpha\, \frac{x-\xi}{\omega}\right)$$

[7] https://github.com/krishgoel/chronocept-baseline-models

Timestamp	Linear (t)	log base 1.1			log base 2			log base 10		
		t'	CR	%	t'	CR	%	t'	CR	%
1 minute	1	0.0	0.000	100	0.0	0.000	100	0.0	0.000	100
1 hour	60	42.96	0.716	28.4	5.91	0.099	90.1	1.78	0.030	97.0
1 day	1440	76.30	0.053	94.7	10.47	0.007	99.3	3.16	0.002	99.8
1 week	10080	96.73	0.010	99.0	13.30	0.001	99.9	4.00	3.968e-4	99.9
1 month	43200	111.97	0.003	99.7	15.39	3.563e-4	99.9	4.63	1.072e-4	~100
1 year	525600	138.23	2.623e-4	~100	19.00	3.615e-5	~100	5.72	1.088e-5	~100
1 decade	5256000	162.25	3.087e-5	~100	22.33	4.249e-6	~100	6.72	1.279e-6	~100

Table 9: Compression analysis across logarithmic bases. CR = t'/t, Compression % = $100 \times (1 - \text{CR})$.

where $\phi(z)$ is the standard normal PDF and $\Phi(z)$ is the standard normal CDF.

Optimization: Parameter estimation is performed using the Trust Region Reflective (TRF) algorithm by minimizing the sum of squared residuals:

$$SSR(\theta) = \sum_{i=1}^{N} (y_i - f(x_i; \theta))^2$$

This is implemented via `scipy.optimize.curve_fit`[8]. After optimization, we compute:

$$N = \int_{x_{\min}}^{x_{\max}} f_{\text{fit}}(x)\, dx,$$

$$f_{\text{norm}}(x) = \frac{f_{\text{fit}}(x)}{N}, \quad f_{\max} = \max_x f_{\text{norm}}(x),$$

$$S_{\text{final}} = \frac{S_{\text{fit}}}{N \cdot f_{\max}}$$

Here, $f_{\text{norm}}(x)$ denotes the AUC-normalized fitted curve corresponding to $p_i(t)$. The additional rescaling captured by S_{final} is applied only when required for visualization or numerical conditioning and is not used in probabilistic interpretation or evaluation.

Evaluation: RMSE is used as the primary goodness-of-fit metric. As a scale-sensitive measure that penalizes large deviations, a lower RMSE indicates superior fit quality.

Table 10 and Figure 5 present the six scenarios, annotation points, and corresponding fitted curves. Table 11 reports RMSE for each candidate distribution across scenarios. The skew-normal consistently yields the lowest RMSE, confirming its suitability for modeling asymmetric and variable-duration temporal profiles.

[8] https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html

E Synthetic Generation of Samples

This section presents the plaintext markdown prompts used for synthetic dataset generation in Chronocept via the GPT-o1 model (OpenAI, 2024). The prompts are designed to yield syntactically coherent text with explicit temporal structure. Generation was performed in batches of 50 samples per prompt.

The prompts are shown in Figure 6 for Benchmark I and Figure 7 for Benchmark II.

F Ablation Study: Impact of Structured Temporal Axes on Model Performance

To evaluate the contribution of multi-axis temporal annotations in modeling temporal validity, we conduct an axis-encoding ablation with a RoBERTa backbone. Specifically, we compare inputs that omit axes entirely, inputs that append inline axis markers in a single sequence, and inputs that concatenate SBERT axis embeddings.

Input Construction. Each Chronocept example is annotated along multiple temporal axes. In the *single_sequence_markers* configuration, axis-specific spans are serialized inline with dedicated markers and concatenated to the parent text as a single sequence. In the *sbert_concat* configuration, SBERT embeddings are computed per axis and concatenated with the parent representation. The *no_axes* control retains only the parent text without any axis structure.

Setup. We compare the three configurations on Benchmark I and Benchmark II using a fixed RoBERTa encoder. Models predict (ξ, ω, α) and are evaluated with MSE, MAE, R^2, NLL, and CRPS.

Results. Table 12 and Table 13 report results and improvements (Δ) relative to the *no_axes*

Temporal Scenario	Sample Sentence	Annotation Points (x, y)
S1: Early Onset	"He is making coffee for himself right now."	(14.91, 0.19), (21.64, 0.41), (27.64, 0.77), (31.64, 0.41), (34.91, 0.20)
S2: Late Onset	"The movie is going to hit the theaters in a few weeks."	(93.75, 0.21), (100.67, 0.80), (106.57, 0.42), (112.73, 0.20), (98.0, 0.63)
S3: Short Duration	"The site has been crashing for a few minutes as there is some server maintenance work going on."	(12.73, 0.21), (28.19, 0.80), (41.28, 0.20), (32.19, 0.60), (18.91, 0.40)
S4: Long Duration	"The ruling government brings growth and progress."	(1, 0.05), (130.38, 0.81), (147.84, 0.21), (111.29, 0.42), (138.38, 0.60)
S5: Rapid Rise, Slow Decay	"The advertisement's impact peaks immediately and lingers."	(42.73, 0.21), (46.91, 0.40), (53.10, 0.80), (63.46, 0.56), (81.83, 0.27)
S6: Slow Rise, Rapid Decay	"The news slowly gains attention but quickly becomes outdated."	(43.28, 0.20), (58.01, 0.40), (76.92, 0.79), (84.92, 0.40), (88.92, 0.17)

Table 10: Six temporal scenarios illustrating the effects of offset, duration, and asymmetry. Each scenario is represented by 5 annotation points on a log-transformed time axis with base 1.1.

control. On Benchmark I, inline markers reduce error modestly and improve calibration, while SBERT concatenation yields the largest gains overall, including a 90.66% reduction in NLL and a 5.95% reduction in MSE. On Benchmark II, *single_sequence_markers* slightly lowers MSE/CRPS but degrades calibration, whereas *sbert_concat* improves R^2 and reduces NLL by 65.30% with a small MSE penalty. These patterns indicate that explicit axis structure improves identifiability, and that embedding-level concatenation can deliver substantial calibration gains when longer contexts are present.

Conclusion. Structured temporal axes improve performance, with the magnitude and locus of gains dependent on input formulation and context length. Inline markers provide robust accuracy improvements; SBERT concatenation delivers the strongest calibration gains on longer inputs. These results validate the use of explicit temporal structure in Chronocept's input design.

G Ablation Study: Impact of Incorrect Temporal Axes Labeling

We evaluate the sensitivity of temporal validity modeling to erroneous axis labeling by conducting an ablation on the RoBERTa baseline. Specifically, we randomly shuffle the order of temporal axes during training while preserving the correct ordering at evaluation.

Setup. Chronocept inputs are constructed by integrating axis-specific content with the parent text. This ablation injects label noise by permuting axis order during training, thereby disrupting the structural alignment between axis semantics and their positions in the sequence. The evaluation set remains unperturbed. Models predict the skew-normal parameters ξ, ω, α and are evaluated on Benchmark II using MSE, MAE, R^2, NLL, and CRPS.

Results. Table 14 shows that shuffling axis order degrades point accuracy and correlation and dramatically worsens calibration. Relative to the correctly ordered control, MSE increases by 0.87%, MAE by 1.09%, R^2 becomes 6.31% more negative,

Distribution	S1	S2	S3	S4	S5	S6	Parameters
Gaussian	0.0709	0.0673	0.0424	0.0273	0.1193	0.0806	$(\mu,\ \sigma)$
Exponential	0.2103	0.2291	0.2312	0.2704	0.2126	0.2212	(λ)
Log-normal	0.0844	0.0597	0.0804	0.0325	0.0872	0.0919	$(\mu,\ \sigma)$
Gamma	0.0827	0.0623	0.0668	0.0307	0.0968	0.0899	$(k,\ \theta)$
Skewed Normal	**0.0514**	**0.0357**	**0.0407**	**0.0224**	**0.0505**	**0.0247**	$(\xi,\ \omega,\ \alpha)$

Table 11: Average RMSE values for candidate distributions across six temporal scenarios. All distributions were fitted using a scaling factor S to enforce AUC $= 1$. A lower RMSE indicates a better fit, as RMSE heavily penalizes large errors due to squaring, is scale-dependent, and more sensitive to outliers.

Benchmark I (RoBERTa)	MSE	MAE	R^2	NLL	CRPS
No Axes	1110.8466	21.3611	-5.1725	1023.1238	52.6468
Single Sequence Markers	1085.3527	21.1018	-5.0833	928.8128	51.9435
Absolute Change (Δ)	+25.4939	+0.2593	+0.0892	+94.3110	+0.7033
Improvement	2.29%	1.21%	1.72%	9.22%	1.34%
SBERT Concatenation	**1044.7926**	**20.0065**	**-3.7696**	**95.5716**	**49.6696**
Absolute Change (Δ)	+66.0540	+1.3546	+1.4029	+927.5522	+2.9772
Improvement	5.95%	6.34%	27.12%	90.66%	5.66%

Table 12: Axis encoding ablation on Benchmark I with RoBERTa. "Absolute Change" and "Improvement" are computed relative to the "No Axes" configuration; positive Δ on R^2 denotes a gain.

and NLL inflates by 358.15%; CRPS increases by 0.44%. These effects indicate that the axis ordering carries non-trivial supervisory signal and that disrupting it injects inductive noise which impairs both fit and uncertainty modeling.

Conclusion. Erroneous axis labeling during training causes statistically meaningful degradation in accuracy and a severe loss of calibration, with NLL increasing by more than a factor of four. Preserving the structural alignment of temporal axes is therefore critical for stable and well-calibrated temporal validity estimation with RoBERTa.

H Ablation Study: Choice of Objective Loss Function for the Baselines

We isolate the effect of the training objective by comparing label-space *MSE*, *Gaussian NLL*, and *Skew-Normal NLL* on DeBERTa-v3, and juxtapose these with a geometry-aware parameter-space Gaussian NLL on MT-DNN.

Unlike location (ξ), scale (ω) and skewness (α) are highly sensitive; small absolute errors translate into large deviations in implied validity curves. Collapsing ξ, ω, α onto a common linear error scale (MSE) is therefore ill-conditioned. Likelihood-based objectives better reflect uncertainty, and the parameter-space Gaussian NLL decouples the geometry by operating in $\xi \in \mathbb{R}$, $\log\omega \in \mathbb{R}$, and $\tilde{\alpha} = \operatorname{artanh}(\alpha/A) \in \mathbb{R}$.

Benchmark I. Label-space MSE on DeBERTa-v3 yields lower squared error than its Gaussian/Skew-NLL counterparts but exhibits severely degraded calibration (NLL $\approx 2.2\times10^2$). Parameter-space Gaussian NLL with MT-DNN attains the strongest overall accuracy with controlled NLL.

Benchmark II. The pattern persists: label-space MSE on DeBERTa-v3 reaches relatively lower squared error than label-space likelihoods yet suffers extreme miscalibration (NLL $\approx 2.71\times10^2$). MT-DNN with parameter-space Gaussian NLL again yields the best aggregate accuracy with stable NLL.

Conclusion. Label-space MSE can reduce average squared error but fails to respect the geometry of ω and α, resulting in severe miscalibration and unreliable rank structure. Label-space likelihoods (Gaussian, Skew-NLL) improve calibration at uneven costs in point error. Geometry-aware parameter-space Gaussian NLL, as instantiated with MT-DNN, delivers the best overall accuracy with stable NLL and should be preferred when reliable estimation of (ξ, ω, α) is required.

Benchmark II (RoBERTa)	MSE	MAE	R^2	NLL	CRPS
No Axes	844.9597	19.0196	-8.5646	1637.6928	47.3738
Single Sequence Markers	**836.9846**	**18.9870**	-8.7325	2944.0318	**47.1757**
Absolute Change (Δ)	+7.9751	+0.0326	-0.1679	-1306.3390	+0.1981
Improvement	0.94%	0.17%	-1.96%	-79.77%	0.42%
SBERT Concatenation	853.9321	18.9265	**-7.9329**	**568.3081**	47.6832
Absolute Change (Δ)	-8.9724	+0.0931	+0.6317	+1069.3847	-0.3094
Improvement	-1.06%	0.49%	7.38%	65.30%	-0.65%

Table 13: Axis encoding ablation on Benchmark II with RoBERTa. "Absolute Change" and "Improvement" are computed relative to the "No Axes" configuration. Negative "Improvement" indicates degradation.

Model	Setting	MSE	MAE	R^2	NLL	CRPS
RoBERTa	Correct Axis Order	**841.7615**	**18.9694**	**-8.4962**	**1494.3527**	**47.3578**
	Shuffled Axis Order	849.0772	19.1761	-9.0324	6846.4384	47.5677
	Absolute Change (Δ)	+7.3157	+0.2067	-0.5362	+5352.0857	+0.2099
	Performance Drop	0.87%	1.09%	6.31%	358.15%	0.44%

Table 14: Axis shuffling ablation on Benchmark II with RoBERTa. “Absolute Change” and “Performance Drop” are computed relative to the correctly ordered control.

Benchmark I	MSE	MAE	R^2	NLL
DeBERTa-v3 (MSE)	**151.0373**	**6.9770**	**-0.0329**	220.4651
DeBERTa-v3 (Gaussian)	695.7505	14.8318	-1.3404	**9.9973**
DeBERTa-v3 (Skew-NLL)	512.7388	13.1731	-1.9905	19.7282
MT-DNN (Skew-NLL)	394.7480	14.3401	-11.2308	4.4120
MT-DNN (Param Gauss)	**108.3768**	**5.9425**	**0.0314**	**4.5117**

Table 15: Objective ablation on Benchmark I, comparing loss function choices for DeBERTa-v3 (label-space) and MT-DNN (parameter-space) training. Lower values are better for all metrics except R^2.

	RMSE			Spearman Coefficient		
Benchmark I	ξ	ω	α	ξ	ω	α
DeBERTa-v3 (MSE)	**20.9545**	**3.5140**	**1.2927**	-0.1261	-0.1255	0.1766
DeBERTa-v3 (Gaussian)	45.5250	3.6120	1.2950	**-0.1179**	**0.0318**	**0.1953**
DeBERTa-v3 (Skew-NLL)	38.7160	5.9390	2.0034	-0.1264	-0.0763	-0.1191
MT-DNN (Skew-NLL)	28.9076	18.4124	3.0951	-0.1122	-0.0578	-0.0740
MT-DNN (Param Gauss)	**17.6030**	**3.6852**	**1.2974**	**0.5200**	**0.1405**	**0.0115**

Table 16: Benchmark I per-parameter RMSE and Spearman.

Benchmark II	MSE	MAE	R^2	NLL
DeBERTa-v3 (MSE)	**182.3851**	**7.4118**	**-0.6490**	270.9766
DeBERTa-v3 (Gaussian)	544.5860	13.7358	-2.7730	**11.6107**
DeBERTa-v3 (Skew-NLL)	674.3253	15.7013	-5.0860	16.5712
MT-DNN (Skew-NLL)	375.5049	15.9036	-32.0211	**4.1012**
MT-DNN (Param Gauss)	**64.5708**	**4.5253**	**-0.0183**	4.1865

Table 17: Objective Loss Function ablation on Benchmark II.

Annotation Guidelines for Chronocept

This document provides instructions for annotating temporal validity using a **three-step process**: **Text Splitting**, **Axis Assignment**, and **Temporal Validity Distribution Plotting**.
These guidelines are tailored to the nature of this benchmark, which typically involves one **Main Axis** segment and one additional axis segment from the seven auxiliary axes.

Step 1: Text Splitting

Objective:
Divide the input sentence into grammatically correct segments, ensuring semantic and temporal integrity is preserved.

Guidelines:
1. **Identify Splitting Points:**
   - Divide the sentence into meaningful subtexts. Most samples will include one **Main Axis** segment and one from the other seven axes.
   - Use punctuation and conjunctions as natural delimiters but ensure that each subtext is self-contained.
2. **Preserve Temporal Context:**
   - Retain essential markers (e.g., *continuously*, *in 2023*, *every month*).
   - Avoid removing or altering any text.
3. **Avoid Over-Splitting:**
   - Ensure each subtext conveys clear, standalone meaning.
   - Over-splitting may lead to fragments that lose context or temporal clarity.
4. **Text Copying Convention:**
   - Copy text exactly as it appears in the sample, including punctuation.
5. **Example:**
   - Input: **"The company is expanding its operations in Asia, and the CEO is leading the efforts, planning a significant increase in market share."**
   - Split:
     - Subtext 1: "The company is expanding its operations in Asia," (Main Axis)
     - Subtext 2: "and the CEO is leading the efforts, planning a significant increase in market share." (Intention Axis)
6. **Ambiguity Handling:**
   - If a sample seems to violate the condition of one Main Axis plus one other axis, document the **Sample ID** and consult **[redacted]**.
   - If a sample does not carry a Main Axis with a clearly definable temporal cue, document the **Sample ID** and consult **[redacted]**.
   - Incorrect samples will be discarded.

Step 2: Axis Assignment

Objective:
Classify each subtext into one of the **seven temporal axes** based on its primary temporal characteristic.
Temporal Axes:
1. **Main Axis (Factual Events):**
   - **Definition**: Verifiable events along a timeline, representing objective truths.
   - **Purpose**: Captures the primary narrative and establishes a concrete temporal sequence.
   - **Example**: "The company is expanding its operations in Asia."
   - **Key Question**: Does this event occur within the primary timeline of the narrative?
2. **Intention Axis:**
   - **Definition**: Captures someone's intention, desire, or plan, even if unfulfilled.
   - **Purpose**: Highlights future-directed actions or goals tied to the narrative but not necessarily realized.
   - **Example**: "The CEO is leading the efforts, planning a significant increase in market share."
   - **Key Question**: Is this event stated as an intended action or goal, regardless of its realization?
3. **Opinion Axis:**
   - **Definition**: Represents subjective viewpoints, expectations, or beliefs about events.
   - **Purpose**: Differentiates opinions or speculations from factual occurrences.
   - **Example**: "Experts believe the market will grow rapidly."
   - **Key Question**: Does this event represent a belief or expectation rather than a verified fact?
4. **Hypothetical Axis:**
   - **Definition**: Includes conditional or hypothetical events dependent on certain conditions.
   - **Purpose**: Tracks scenarios that are imagined or conditional, often using "if" statements.
   - **Example**: "If the company secures funding, it will expand globally."
   - **Key Question**: Is this event presented as dependent on another event or condition?
5. **Negation Axis:**
   - **Definition**: Identifies events explicitly stated as not occurring.
   - **Purpose**: Tracks denied actions or outcomes to separate them from realized events.
   - **Example**: "The company did not expand its operations in 2020."
   - **Key Question**: Is this event explicitly stated as unfulfilled or negated?
6. **Generic Axis**:
   - **Definition**: Represents universal truths or habitual occurrences, not tied to a specific timeline.
   - **Purpose**: Highlights timeless facts or generalizations applicable broadly.
   - **Example**: "Lions eat meat."
   - **Key Question**: Is this event a universal truth or a habitual occurrence that transcends specific contexts?
7. **Static Axis:**
   - **Definition**: Captures unchanging states or conditions **within a specific context or timeframe**.
   - **Purpose**: Tracks context-dependent facts or conditions relevant to the narrative.
   - **Example**: "The room is cold."
   - **Key Question**: Is this event context-specific and static within the described situation?
8. **Recurrent Axis:**
   - **Definition**: Describes events or states that happen repeatedly over time.
   - **Purpose**: Tracks patterns or cycles of actions/events relevant to the narrative.
   - **Example**: "The train arrives every morning at 8 AM."
   - **Key Question**: Does this event represent a recurring action or pattern?

Guidelines:
1. **Assign to the Closest Axis:**
   - Carefully analyze the temporal and semantic meaning of the subtext.
   - Decide if the event can be anchored to a specific axis based on its nature.
   - Most samples will have one **Main Axis** subtext and one auxiliary axis subtext.
2. **Handle Ambiguities:**
   - Focus on the start-points of events to reduce ambiguity related to durations.
   - Only compare events on the same axis; cross-axis relations require separate investigation.
   - If unsure about the axis, document the **Sample ID** and consult **[redacted]**.
   - Incorrect samples will be removed from the dataset.
3. **Use Context:**
   - Assess the broader context to distinguish between axes like Static and Generic.
4. **Example Annotation:**
   - Subtext: "The CEO is leading the efforts, planning a significant increase in market share."
     - Assigned Axis: **Intention Axis**
5. **Advisory for Complex Cases:**
   - Consider the following example: "The printer is making strange noises while the IT technician tries to fix it."
     - "The IT technician is trying to fix the printer" can be treated as the **Main Axis**, while "the printer is making strange noises" can be assigned to the **Generic Axis**.
     - This requires thoughtful analysis, as the roles of subtexts may not be apparent immediately. Annotators should carefully consider such cases, akin to transposing the segments for clarity.

Step 3: Temporal Validity Distribution Plotting

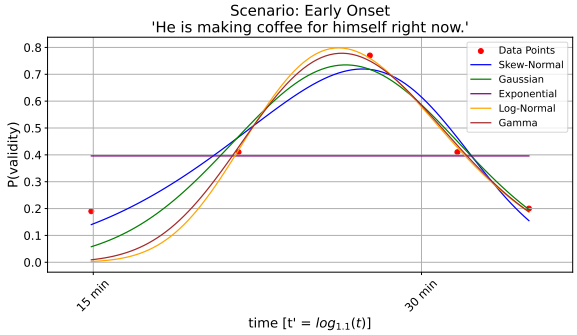
Objective:
Plot a skewed probability distribution over a **time graph** to represent the temporal validity of each subtext.

Guidelines:
1. **Temporal Cue Assignment:**
   - For samples with clear temporal cues (e.g., "solving for 1 hour"), assign a time interval to that cue. As an advisory, consider that a vernacularly used "1 hour" can range from 45 minutes to 90 minutes.
2. **Graph Axes:**
   - **X-Axis (Time):**
     - Labeled with intervals: 1 minute, 15 minutes, 30 minutes, 1 hour, 12 hours, 1 day, 1 week, 1 month, 1 year, 1 decade, and infinite validity.
   - **Y-Axis (Probability):**
     - Range: 0 (not valid) to 1 (fully valid).
3. **Plotting Points:**
   - Place 3–5 points on the timeline to indicate the probability of validity at specific times.
   - The user need not worry about making an ideal probability distribution with **AUC = 1**. Instead, plot proportions relative to the temporal "point" with the highest probability (Maximum Likelihood Estimate, MLE).
4. **Fit a Skewed Probability Distribution:**
   - A skewed curve will be automatically fitted through the plotted points to represent the temporal validity distribution.
5. **Consistency:**
   - Maintain consistency in plotting for similar subtexts.
6. **Ambiguity Handling:**
   - If the sample is technically correct but you are highly unsure about the temporal interval, annotate to the best of your ability. Low inter-annotator agreement (IAA) samples will be flagged and eliminated
during post-processing.
   - If unsure about the distribution, document the **Sample ID** and consult **[redacted]**.
   - Incorrect samples will be removed.
The result of this step is a skewed probability distribution reflecting the temporal validity over time.

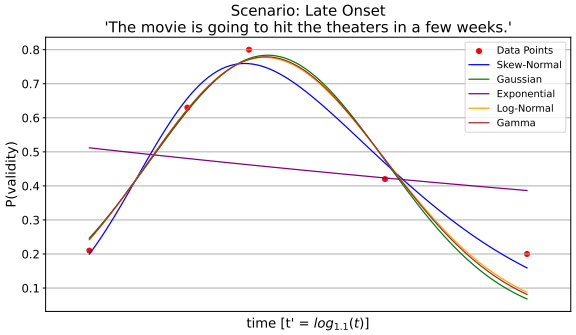
General Notes for Annotators

1. **Ambiguities:**
   - For unclear splits, axis assignments, or validity distributions, contact **[redacted]** with the **Sample ID** for resolution.
2. **Discarding Samples:**
   - Multimodal samples or those with excessive ambiguity should be flagged for review and potential removal.
3. **Temporal Objectivity:**
   - Avoid consulting peers during annotation to maintain objectivity and ensure consistency across annotators.
4. **Quality Control:**
   - Ensure all annotations are thorough, consistent, and adhere to these guidelines.

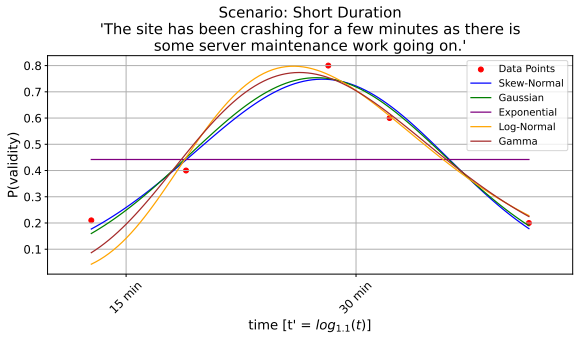
Figure 2: Annotation guidelines for Chronocept.



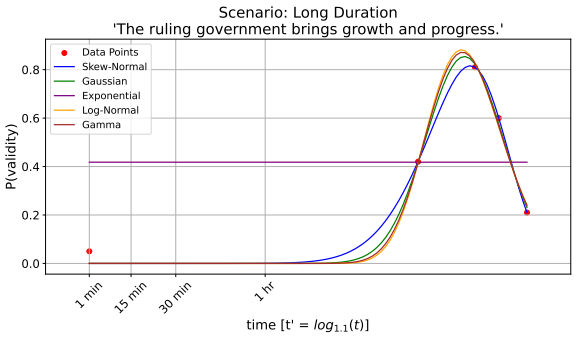
(a) Early Onset: Peak validity occurs soon after publication.



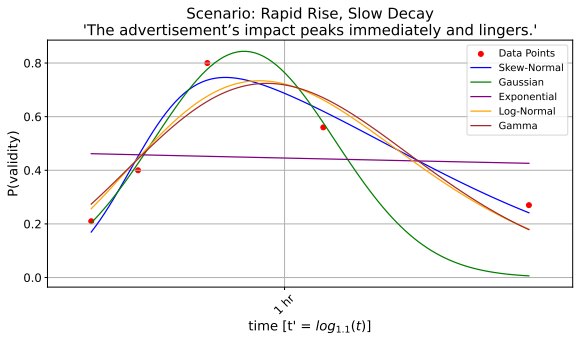
(b) Late Onset: Validity emerges gradually and peaks later.



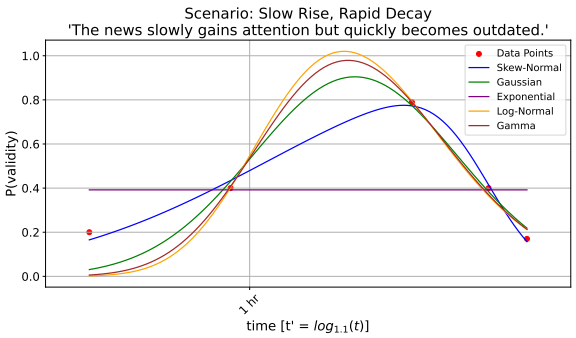
(c) Short Duration: A narrow window of high relevance.



(d) Long Duration: Validity persists over time.



(e) Rapid Rise, Slow Decay: Sudden onset, gradual decline.



(f) Slow Rise, Rapid Decay: Gradual onset, sharp drop.

Figure 5: Visual fit comparison of candidate distributions across six temporal scenarios. The skew-normal consistently provides the best fit, modeling varied validity patterns in onset, duration, and asymmetry.

	RMSE			Spearman Coefficient		
Benchmark II	ξ	ω	α	ξ	ω	α
DeBERTa-v3 (MSE)	**23.2404**	**2.4431**	**1.0349**	-0.2178	0.1617	**0.0159**
DeBERTa-v3 (Gaussian)	40.2941	2.9834	1.1133	**-0.1996**	**0.3994**	0.0037
DeBERTa-v3 (Skew-NLL)	44.8280	2.5648	2.6169	-0.2608	0.0321	0.0090
MT-DNN (Skew-NLL)	26.8896	19.3944	5.2273	0.0603	-0.0872	**0.1009**
MT-DNN (Param Gauss)	**13.6704**	**2.4017**	**1.0319**	**0.1807**	-0.0488	0.0042

Table 18: Benchmark II per-parameter RMSE and Spearman.
0

Synthetic Data Generation for a Temporal Validity Benchmark

Objective
This task involves creating synthetic sentences that will form the basis of a benchmark for temporal validity research. Your role as a text generation model is to produce *high-quality sentences only*, without accompanying explanations or axis descriptions. These sentences should describe occurrences or events that happen simultaneously or contrastively, incorporating various actions, states, or processes.

Key Definition: Axis
An axis represents a semantic dimension or characteristic used to classify and analyze the relationships between events in a sentence. Axes are categorized into two types:
1. **Event-Related Axes**: Describe the relationship between events or states in a sentence, focusing on interactions or dependencies.
2. **Annotation Axes**: Provide supplementary semantic information about the events, enhancing interpretability.

Event-Related Axes
Specify the relationship between events in the sentence:
1. **Temporal Overlap**: Events occur simultaneously or in parallel.
2. **Causality**: One event causes or results from the other.
3. **Subordination**: One event depends on or occurs due to the other.
4. **Unrelated**: Events are independent of each other.

Annotation Axes
Provide semantic context and additional dimensions of meaning:
1. **Main Axis (Factual Events)**: Verifiable, objective events tied to a specific timeline.
2. **Intention**: Future-directed plans, desires, or actions.
3. **Opinion**: Subjective beliefs or expectations about events.
4. **Hypothetical**: Conditional or imagined scenarios.
5. **Negation**: Explicitly unfulfilled or denied actions or outcomes.
6. **Generic**: Universal truths or habitual actions that apply broadly across contexts and are not tied to specific timelines.
7. **Static**: Unchanging states or conditions that are specific to a particular context or timeframe.
8. **Recurrent**: Events or states that recur over time, forming patterns or cycles.

Guidelines for Sentence Generation

Sentence Structure
- Sentences should be written in the *present tense*. Use ***all forms of present tense*** - Simple Present Tense, Present Continuous Tense, Present Perfect Tense and Present Perfect Continuous Tense.
- Each sentence should incorporate:
  - *At least one Event-Related Axis* to define the relationship between events.
  - *Two Annotation Axes, one of which must be the ***Main Axis (Factual Events)****.

Neutrality and Diversity
- Sentences must span *diverse domains*, including daily life, technology, abstract concepts, and nature.
- Use a mix of *pronouns* ("he," "she," "they"), *generic entities* (e.g., "a person," "a machine"), and *articles* ("the," "a"). Ensure pronouns are evenly distributed across the dataset to represent diverse actors.

Task Output
1. Generate *50 sentences* adhering strictly to the above structure and requirements.
2. Ensure diversity in domains, axes, and event relationships while maintaining clarity and coherence.
3. Each sentence must explicitly include:
   - ***At least one Event-Related Axis***.
   - ***Two Annotation Axes***, with the *Main Axis (Factual Events)* included.

Examples of Correct Sentences
1. "She is cooking dinner, but the oven keeps malfunctioning."
2. "He is driving to work, while the traffic jam is worsening."
3. "They are reviewing documents, as the deadline approaches."
4. "A researcher is designing an experiment, while the technician prepares the equipment."
5. "The sky is darkening, but the lake remains calm and still."
6. "A student is reading the manual to understand how the device might operate."
7. "She is negotiating a contract, while her team finalizes the presentation."
8. "The clouds are gathering, and the wind is picking up speed."
9. "The robot is performing a task, while the operator monitors its efficiency."
10. "He is practicing the piano, but the audience remains silent."

Figure 6: Plaintext markdown prompt for Benchmark I.

Synthetic Data Generation for a Temporal Validity Benchmark

Objective
Your role as a text generation model is to produce *high-quality, coherent, and naturally flowing sentences or short paragraphs*, without accompanying explanations or axis descriptions. These samples should describe occurrences or events that happen simultaneously or contrastively, incorporating various actions, states, or processes. Avoid unnatural, overly formal, or stilted constructions.

Key Definition: Axis
An axis represents a semantic dimension or characteristic used to classify and analyze the relationships between events in a sentence. Axes are categorized into two types:
1. **Event-Related Axes**: Describe the relationship between events or states in a sentence, focusing on interactions or dependencies.
2. **Annotation Axes**: Provide supplementary semantic information about the events, enhancing interpretability.

Event-Related Axes
Specify the relationship between events in the sentence:
1. **Temporal Overlap**: Events occur simultaneously or in parallel.
2. **Causality**: One event causes or results from the other.
3. **Subordination**: One event depends on or occurs due to the other.
4. **Unrelated**: Events are independent of each other.

Annotation Axes
Provide semantic context and additional dimensions of meaning:
1. **Main Axis (Factual Events)**: Verifiable, objective events tied to a specific timeline.
2. **Intention**: Future-directed plans, desires, or actions.
3. **Opinion**: Subjective beliefs or expectations about events.
4. **Hypothetical**: Conditional or imagined scenarios.
5. **Negation**: Explicitly unfulfilled or denied actions or outcomes.
6. **Generic**: Universal truths or habitual actions that apply broadly across contexts and are not tied to specific timelines.
7. **Static**: Unchanging states or conditions that are specific to a particular context or timeframe.
8. **Recurrent**: Events or states that recur over time, forming patterns or cycles.

Guidelines for Sentence Generation

Sentence Structure
- Sentences should be written in the *present tense*. Use **all forms of present tense** - Simple Present Tense, Present Continuous Tense, Present Perfect Tense and Present Perfect Continuous Tense.
- Each sentence should incorporate:
  - *At least two Event-Related Axes* to define the relationship between events.
  - *Four or more Annotation Axes*, one of which must be the **Main Axis (Factual Events)**.
- Avoid overusing commas. Instead, use full stops to separate ideas into distinct sentences where appropriate.

Neutrality and Diversity
- Sentences must span *diverse domains*, including daily life, technology, abstract concepts, and nature.
- Use a mix of *pronouns* ("he," "she," "they"), *generic entities* (e.g., "a person," "a machine"), and *articles* ("the," "a"). Ensure pronouns are evenly distributed across the dataset to represent diverse actors.

Task Output
1. Generate *50 sentences* adhering strictly to the above structure and requirements.
2. Ensure diversity in domains, axes, and event relationships while maintaining clarity and coherence.
3. Each sentence must explicitly include:
   - **At least two Event-Related Axis**.
   - **Four or more Annotation Axes**, with the *Main Axis (Factual Events)* included.

Examples of Correct Sentences
1. “She is cooking dinner. At the same time, the oven is malfunctioning, which causes delays in her preparation. She checks the ingredients repeatedly, ensuring nothing is missing, while worrying that the dish may not turn out as planned. Despite the challenges, she intends to serve the meal on time to surprise her family.”
2. “He is driving to work, navigating through dense traffic as the morning rush intensifies. Meanwhile, the traffic jam worsens due to a nearby accident, forcing him to rethink his route while calculating the estimated delay. He considers taking a detour through side streets, hoping to save time, but worries it might lead to further complications.”
3. “She is watering the garden while the sun remains hidden behind the clouds, leading to slower evaporation. She frequently checks the soil moisture, believing that overwatering might damage the plants, though she intends to use organic fertilizer soon.”

Figure 7: Plaintext markdown prompt for Benchmark II.