Evaluating Multi-Dimensional Generalization of Large Language Models in Temporal Extraction Tasks

Fahmid Shahriar Iqbal[1], **Ritam Dutt**[2], **Soumitra Das**[3], **Arnav Verma**[4], **Sagnik Ray Choudhury**[1]

[1]University of North Texas, [2]Carnegie Mellon University
[3]University of Massachusetts Amherst, [4]New York University

fahmidshahriar.iqbal@unt.edu, rdutt@andrew.cmu.edu, soumitradas@umass.edu
av4445@nyu.edu, sagnik.raychoudhury@unt.edu

Abstract

Time and event expression extraction are fundamental temporal reasoning tasks, but the problem remains difficult due to annotation ambiguity, domain sensitivity, and unstable model behavior. Existing evaluations focus on in-domain performance, offering limited insight into reliability under distribution shifts. We evaluate multiple model configurations across families, architectures, and reasoning strategies over four dimensions of generalization, examining transfer from base performance, cross-dimensional correlations, and the effects of scale, architecture, and prompting. This provides a systematic study of how prompted LLMs generalize in time and event expression extraction tasks. We find that strong base-task performance generally predicts better generalization. However, this relationship weakens under substantial distribution shifts. Inductive prompting performs most consistently across domain shift, adversarial perturbations, compositionality, and length increase, while gains from scale, architecture, and deductive and abductive prompting strategies are uneven and dimension-specific. We conclude that LLM generalization in temporal extraction tasks cannot be predicted from any single dimension alone and cannot be reliably inferred from in-domain or single-dimension evaluations, highlighting the need for reasoning strategies that generalize across dimensions.

1 Introduction

Temporal reasoning in NLP aims to infer temporal structure from text, including events, time expressions, and the relations that order them over time (Pustejovsky et al., 2003; Ning et al., 2018). Time and event expression extraction is a necessary first step in that process: in the sentence "Monday the company announced a merger," a system must identify "Monday" and "announced" before determining when the announcement occurred.

Low inter-annotator agreement (Cassidy et al., 2014; Styler IV et al., 2014) in such span-level annotation points to the inherent difficulty of the problem. LLMs are increasingly applied to this problem, but recent work shows they are often inconsistent and unreliable (Qiu et al., 2024; Kougia et al., 2024). Consequently, robust extraction remains a critical bottleneck for reliable temporal reasoning, and in-domain evaluation alone is insufficient to assess reliability. Instead, models must be tested for generalizability.

Prior work argues for testing models in multi-dimensional generalization settings, covering different types of distributional shift and compositional variation (Hupkes et al., 2023). Such multi-dimensional generalization is crucial in temporal extraction, where deployed models must handle domain shifts, perturbations, and compositional complexity (Dutt et al., 2024).

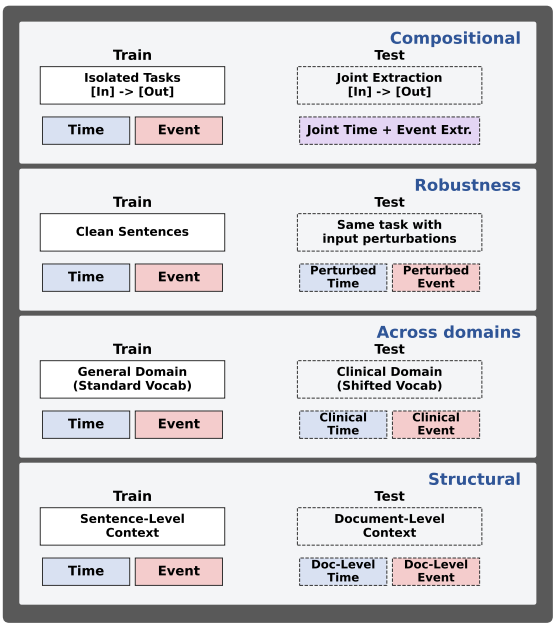


Figure 1: Generalization Dimensions.

Code: GitHub Datasets: HF Models: HF

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 605–629
November 6-10, 2026 ©2026 Association for Computational Linguistics

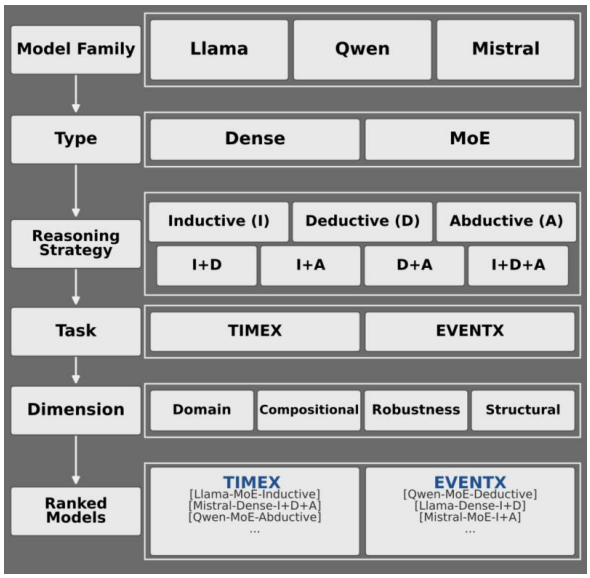


Figure 2: Pipeline for evaluation across tasks, architectures, and reasoning strategies.

We present a systematic study of the multi-dimensional generalization of recent LLMs in time and event expression extraction tasks (TIMEX and EVENTX). Following prior work (Hupkes et al., 2023; Dutt et al., 2024), we consider four dimensions of generalization (Figure 1). **Domain generalization** measures models' ability to handle distribution shifts across domains with different linguistic characteristics (e.g., general vs. clinical text). **Robustness generalization** tests models' robustness to meaning-preserving perturbations. For **Compositional generalization**, the task is extended from extracting time and event expressions independently to extracting them jointly. Finally, for **Structural generalization**, the input context varies from sentence-level to document-level. These dimensions form one axis of variation, capturing how model behavior changes under different types of distribution shifts.

In parallel, we consider reasoning strategies as another axis of variation in model behavior. Reasoning strategies define how inputs are framed and structured under different prompting conditions, including deductive, inductive, and abductive reasoning, as well as combinations of these (Liu et al., 2024), and performance under one strategy does not necessarily transfer to others. Together, multiple generalization dimensions and reasoning strategies define two complementary axes that enable a systematic evaluation of model configurations.

We evaluate models from three families: LLaMA (Touvron et al., 2023), Qwen (Bai et al., 2023), and Mistral (Jiang et al., 2024), and for each, we use 2 architectural variants: Dense and Mixture-of-Experts (MoE). This, combined with the 7 reasoning strategies, yields a total of 42 model configurations ($\mathcal{M}_i$s). Each $\mathcal{M}_i$ is evaluated on both temporal and event extraction tasks across all generalization dimensions. With this extensive setup (Figure 2), we ask the following research questions:

- **RQ1:** If a model configuration $\mathcal{M}_i$ performs well on base extraction tasks, does the performance consistently extend across all generalization dimensions? If not, practitioners risk selecting $\mathcal{M}_i$s that perform well in standard evaluations but fail in practice (§3.1).
- **RQ2:** To what extent are different generalization dimensions correlated? Specifically, if a configuration performs well in one dimension, how likely is it to perform well in others? Unlike RQ1, which studies transfer from base tasks, RQ2 examines interdependence among generalization dimensions (§3.2).
- **RQ3:** How do prompting and model factors affect generalization across dimensions? We examine the effects of reasoning strategy, model architecture (Dense vs. MoE), and model size. These factors are orthogonal to the generalization dimensions studied in RQ1 and RQ2, and their effects may vary across dimensions (§3.3).

We show that strong base-task performance generally predicts multi-dimensional generalization, except under substantial distribution shifts. Among reasoning strategies, inductive ones perform well on base tasks as expected, but they also exhibit strong multi-dimensional generalization and greater cross-dimensional consistency. Surprisingly, combining instructions (deductive) with demonstrations (inductive) does not meaningfully improve generalization, whereas prompting the model to produce explanations (abductive) does. MoE models outperform their dense counterparts on base tasks and across many dimensions of generalization, but this advantage diminishes under substantial distributional shifts. Finally, scaling model size improves adversarial and structural generalization but yields no significant gains for domain or compositional ones. Compared to prompted LLMs, fine-tuned PLMs used as baselines exhibit more consistent cross-dimensional behavior, though this consistency does not always reflect strong perfor-

mance. These findings call for reasoning strategies and evaluation frameworks that explicitly target robust, cross-dimensional generalization rather than assuming it follows from strong base-task performance.

2 Experimental Setup

2.1 Tasks & Datasets

Temporal reasoning in natural language requires identifying what happened and when. This involves extracting event expressions (EVENTX), which capture semantic occurrences (e.g., actions, states, or reporting events), and temporal expressions (TIMEX), which capture dates, times, durations, or relative references like today or last week. These span extraction problems are prerequisites for advanced temporal reasoning tasks such as ordering events, determining causality, or building timelines. Therefore, we focus on them.

Base Corpus: TimeBank. We derive the base datasets for both TIMEX and EVENTX tasks from the TimeBank 1.2 corpus (Pustejovsky et al., 2003). The corpus is converted into sentence-level instances by segmenting documents and retaining only sentences that contain at least one target expression, with annotations projected accordingly. We remove all additional annotation attributes (e.g., normalization for temporal expressions and linguistic features for events). Each dataset consists of sentence–target pairs, with a raw sentence as input and a set of textual spans as output.

Adversarial Datasets: *Adv* & *Adv_P*. We construct adversarial variants of the base datasets using a modified SeqAttack framework (Simoncini and Spanakis, 2021), adopting two widely used attack strategies: TextFooler (Jin et al., 2020) and DeepWordBug (Gao et al., 2018). TextFooler is a word-level attack that iteratively replaces important tokens with semantically similar synonyms while preserving the overall meaning. In contrast, DeepWordBug operates at the character level, introducing small perturbations (e.g., swaps, insertions, deletions) that disrupt tokenization and degrade model performance. For TIMEX we use TextFooler and DeepWordBug to generate adversarial examples for each sentence and merge the perturbations. For EVENTX, we restrict the perturbation to TextFooler, as event triggers are sensitive to local syntactic context. These datasets are referred to as *Adv*.

We also construct a prompt-based adversarial variant (*Adv_P*) using GPT-4.1(Xu et al., 2023). We prompt the model to introduce lexical substitutions, character-level noise, and structural modifications, while preserving sentence meaning. We validate that all target expressions remain unchanged.

Domain-Shift: Clinical Dataset (*T2C*) and Vocabulary Replacement Dataset (*Voc*). To evaluate cross-domain generalization, we construct datasets (*T2C*) using the 2012 i2b2 Clinical Temporal Relations corpus (Sun et al., 2013). Compared to TimeBank news data, clinical narratives contain specialized medical vocabulary, abbreviations, and telegraphic sentences. To ensure consistency, we apply the same preprocessing pipeline as in the base datasets.

We create another domain dataset (*Voc*) by systematically replacing selected tokens (e.g. entity-like tokens, rare words, and domain-specific terms) with synthetic or semantically related alternatives, preserving sentence structure and label (Ribeiro et al., 2020). Compared to *T2C*, *Voc* represents a more controlled and localized shift in vocabulary rather than a full domain change.

Structural Generalization: Document Level Dataset (*Len*). Structural generalization refers to models' ability to effectively handle inputs of varying or increased length. We revert to the original documents of the TimeBank corpus for this purpose.

Compositional Dataset (*Comp*). Compositional generalization captures a model's ability to solve more complex tasks by systematically combining solutions to smaller, simpler sub-tasks it has learned. To mimic this, we repurpose the TimeBank dataset so that the models must jointly extract time and event expressions.

See §A.4 for a detailed discussion, and examples from all base and generalization datasets (Figures 20 and 21).

2.2 Models

Based on prior work highlighting trade-offs between throughput, latency, and memory efficiency during inference (Agrawal et al., 2024; Fu et al., 2024), we organize models into two computational tiers.

Medium Tier (Dense Architectures, 16–24 GB VRAM) consists of compact dense models that run efficiently on a single GPU. We include Mistral 8B, Llama 3 8B Instruct, and Qwen2.5 7B Instruct, representing small-scale dense architectures with comparable parameter sizes.

Heavy Tier (Mixture-of-Experts Models, $\geq$ 48 GB VRAM) includes large-scale mixture-of-experts (MoE) models with conditional routing, enabling us to study the effect of expert activation on generalization. We evaluate Llama 4 Maverick (17B active parameters), Qwen3 235B A22B (22B active parameters), and Mixtral 8x7B Instruct (13B active parameters).

Scaling. We conduct a controlled study on the Mistral dense model family: Mistral 3B, Mistral Small 3.2 (24B), and Mistral Large (123B), to examine how increasing parameter count impacts generalization.

2.3 Reasoning strategy

Reasoning strategy (Liu et al., 2024) forms one of the primary axes of our study, alongside the different generalization dimensions. We evaluate each model under seven strategies: inductive ($\mathcal{I}$), deductive ($\mathcal{D}$), abductive ($\mathcal{A}$), and their combinations, across both TIMEX and EVENTX tasks.

Inductive reasoning corresponds to in-context learning, in which the task is implicitly specified by examples from which the model infers patterns (Brown et al., 2020). We implement this via few-shot prompting (§ A.3.1), where models infer extraction patterns from annotated examples. For consistency, examples are always drawn from the base dataset, and the best-performing number of exemplars ($k \in 3, 5, 8$) is fixed across all settings. In contrast, the **deductive** reasoning does not use examples; instead, the task is explicitly specified as a set of rules or a natural language instruction, and the model applies these rules to produce outputs. Specifically, we provide a clear definition of valid time and event expressions (§ A.3.2). **Abductive** reasoning involves generating intermediate hypotheses or explanations to guide prediction (Wei et al., 2022). In our setup, we implement this by generating candidate spans with reasoning steps and selecting those that best justify the extraction (§ A.3.3). We also consider combinations of reasoning strategies that integrate multiple paradigms within a single prompt. For instance, $\mathcal{D} + \mathcal{I}$ incorporates both instructions and few-shot examples, enabling the model to align explicit rules with observed patterns. These hybrid setups capture interactions between reasoning mechanisms that are not observable in isolated settings.

2.4 Evaluation Metrics

We evaluate the quality of predicted expressions using a Hungarian word-level matching metric and report the micro F1 score as our primary metric.

Let $\mathcal{P} = \{p_1, \ldots, p_m\}$ denote the set of predicted expressions and $\mathcal{G} = \{g_1, \ldots, g_n\}$ denote the set of gold expressions for a given instance. For each pair (p_i, g_j), we compute a similarity score based on the word-level Longest Common Subsequence (LCS), normalized by span length: $\text{sim}(p_i, g_j) = \frac{\text{LCS}(p_i, g_j)}{\max(|p_i|, |g_j|)}$ where $|p_i|$ and $|g_j|$ denote the number of tokens in the spans.

We construct a similarity matrix $\mathbf{S} \in \mathbb{R}^{m \times n}$ with entries $S_{ij} = \text{sim}(p_i, g_j)$ and compute an optimal one-to-one matching $\mathcal{M} \subseteq \mathcal{P} \times \mathcal{G}$ that maximizes the total similarity: $\mathcal{M} = \arg\max_{\text{matching}} \sum_{(i,j)\in\mathcal{M}} S_{ij}$ using the Hungarian algorithm (Kuhn, 1955). Precision $\frac{\sum_{(i,j)\in\mathcal{M}} S_{ij}}{|\mathcal{P}|}$ and Recall $\frac{\sum_{(i,j)\in\mathcal{M}} S_{ij}}{|\mathcal{G}|}$ are then defined based on the matched overlaps.

This approach prevents multiple matches to the same expression, providing a more rigorous evaluation similar to span-based protocols used in the CoNLL-2003 Shared Task (Sang and De Meulder, 2003) and SemEval-2013 Task 9.1 (Segura-Bedmar et al., 2013).

3 Results

All prompted LLMs are accessed via the OpenRouter API[1] with default settings and temperature 0, ensuring a consistent and reproducible inference setup. The detailed results are provided in §A.5.

3.1 RQ1: Transferability of Base Task Performance Across Dimensions

We evaluate whether strong performance on base extraction tasks translates across all generalization dimensions. To do so, we examine whether pairwise orderings between configurations, established on the base dataset, persist across generalization dimensions. Specifically, consider two configurations $\mathcal{M}_i$ and $\mathcal{M}_j$ where i is significantly better than j on the *base* dataset. What is the probability that $\mathcal{M}_i$ will be significantly better, worse, or almost equal to $\mathcal{M}_j$ in *generalization* datasets? Figure 3 shows the results for both TIMEX and EVENTX datasets. Statistical significance ($p < 0.05$) is determined using a paired bootstrap test (Berg-Kirkpatrick et al., 2012).

[1] https://openrouter.ai/

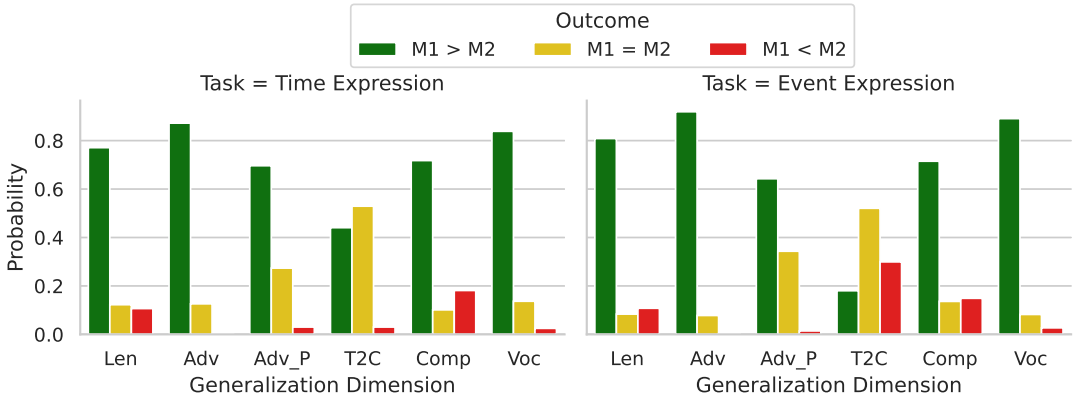


Figure 3: If $\mathcal{M}_1$ is significantly better than $\mathcal{M}_2$ on the base case, what is the probability that the advantage will hold for different generalization dimensions?

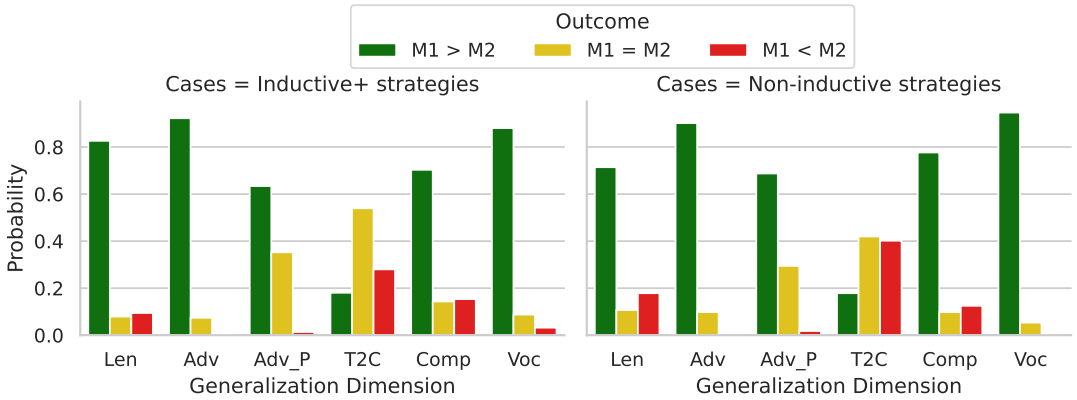


Figure 4: Generalization probabilities for EVENTX, split into inductive+ (left) and non-inductive (right) strategies.

First, no configuration achieves a 100% probability. Based on base-task performance, no single $\mathcal{M}_i$ consistently guarantees better generalization than the others. However, for TIMEX, if an $\mathcal{M}_i$ performs better than $\mathcal{M}_j$ on the base task, it is highly likely to generalize better than $\mathcal{M}_j$ on adversarial, length, and compositional datasets. If the domain generalization dataset has a very strong distribution shift (e.g., *T2C*), then the models will have a much lower probability of generalizing; otherwise, e.g., with a simple vocabulary shift (*Voc*), the generalization trend will hold. EVENTX results are also similar, but the *T2C* generalization probability reduces even further.

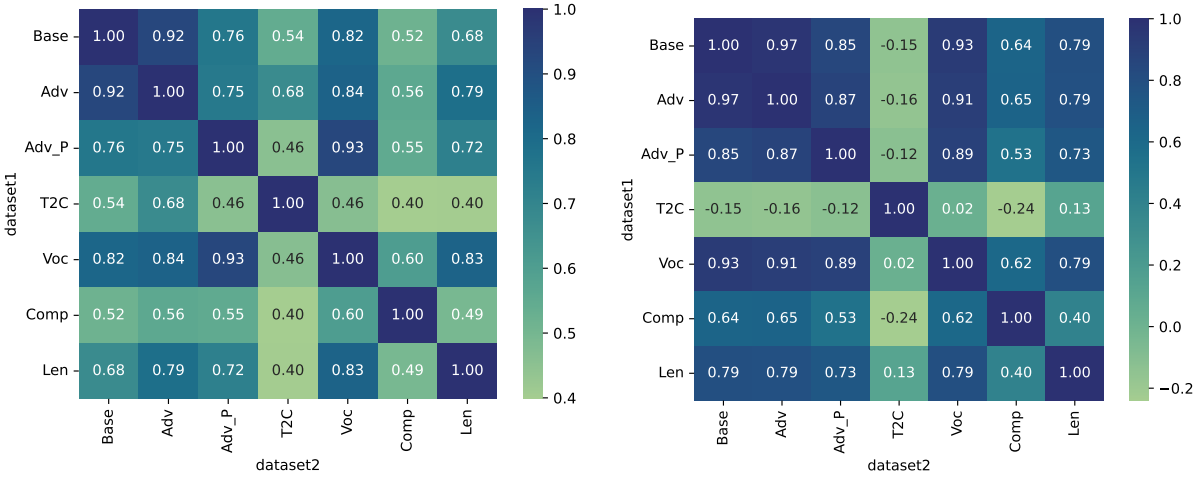
Can the *T2C* anomaly be attributed to *inductive+* ($\mathcal{I}$, $\mathcal{A}+\mathcal{I}$, $\mathcal{D}+\mathcal{I}$, $\mathcal{A}+\mathcal{D}+\mathcal{I}$) reasoning strategies? These strategies generally yield better performance on the base dataset (see Figure 12, §A.1.1), likely because they use exemplars drawn from the same distribution. However, this use of in-domain examples may negatively affect generalization. In Figure 4 we plot these probabilities separately for inductive+ and non-inductive strategies for EVENTX datasets. Interestingly, the use of in-domain examples boosts generalization, even for difficult domain shifts such as *T2C*.

3.2 RQ2: Correlation Across Generalization Dimensions

For RQ2, we assess whether configurations ($\mathcal{M}_i$) maintain consistent relative performance across different generalization dimensions. We first rank all configurations separately on each dataset, then compute Spearman's rank correlation (ρ) between every pair of datasets to measure how similar these rankings are. Figures 5a and 5b show the results for TIMEX and EVENTX tasks, respectively.

For TIMEX datasets, model rankings show moderate-to-strong consistency across most generalization dimensions. Adversarial perturbation datasets are well correlated with each other as expected.[2] The domain-shift datasets (*T2C* and *Voc*) have lower but moderate correlations. Interestingly, adversarial generalization (across both datasets) is

[2] $\rho = 0.75$, and $\rho \geq 0.6$ indicates strong to very-strong correlation (Statstutor, 2004).



(a) Time expression datasets

(b) Event expression datasets

Figure 5: Spearman's Rank Correlation ρ between the source and the target datasets for different generalization dimensions: temporal and event expression extraction.

well correlated with domain and structural generalization, but not with compositional generalization. These trends tend to hold for event expression datasets, except for *T2C*, which is, in fact, negatively correlated with other dimensions. A negative correlation implies that $\mathcal{M}_i$s that perform well on *T2C* tend to perform worse on other dimensions, i.e., optimizing for this domain shift may harm generalization elsewhere. As before, we investigate whether this is due to sample selection in the inductive strategies. Figures 13a and 13b (§A.1.2) show that this is not the case. The correlations, although low, are much higher in the inductive+ strategies than the non-inductive ones.

To further characterize *T2C* 's distinct behavior, we analyze its extraction errors in Figure 6. For TIMEX, errors consist of both missing and boundary-mismatched spans, whereas EVENTX errors are dominated by missing spans, with 66.7–79.5% of gold event expressions missed across models. The t-SNE projections in Figure 7, based on final-layer `roberta-base` embeddings of the 100 most frequent tokens, further illustrate the distributional differences among Base, *T2C*, and *Voc*. Greater separation is observed between *T2C* and Base tokens than between *Voc* and Base, suggesting a stronger lexical and contextual shift. We hypothesize this to be a reason for *T2C* behaving differently from other generalization dimensions.

In summary, a strong base-task performance of $\mathcal{M}_i$ is generally a good indicator of multi-dimensional generalization, with some exceptions such as significant distribution shifts (e.g., *T2C*). Most importantly, inductive+ strategies, where few-shot examples are sampled from the base datasets, not only boost *base performance* (Figure 12, §A.1.1) but also improve generalization. Furthermore, they maintain significantly higher cross-dimensional correlation than non-inductive approaches.

3.3 RQ3: Effects of Reasoning Strategies, Model Architecture, and Scale

Reasoning strategy. The previous results indicate that inductive+ strategies are better for generalization. Following up, we investigate whether, among these strategies, a particular one, say, $\mathcal{D}+\mathcal{I}$, is better than the others. Specifically, for any reasoning strategy pair (r_i, r_j) and a generalization dimension, say, *T2C*, we note the percentage of models that perform significantly ($p < 0.05$) better, worse, or almost equal ($p \geq 0.05$) for r_i than r_j. Figure 8 shows the results when the results are averaged over *all dimensions*. Results for individual dimensions are shown in Figures 14 and 15, §A.1.3, and have similar trends.

For TIMEX, addition of deductive prompts, i.e., instructions, hampers generalization, but the addition of explanations improves it ($\mathcal{I}$ is better than $\mathcal{I}+\mathcal{D}$, and $\mathcal{I}+\mathcal{A}$ is better than $\mathcal{I}+\mathcal{D}$). Perhaps most interestingly, $\mathcal{I}+\mathcal{D}+\mathcal{A}$, i.e., "few-shot chain-of-thought" is not the best reasoning strategy for generalization. For EVENTX, the base inductive strategy ($\mathcal{I}$) is better than all other alternatives.

Model architecture. For each generalization dimension, we report whether Dense models perform

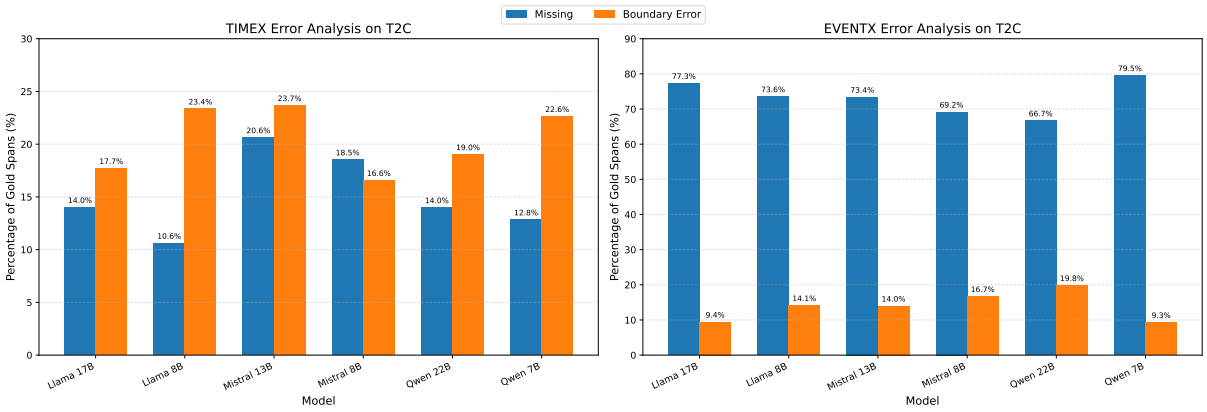
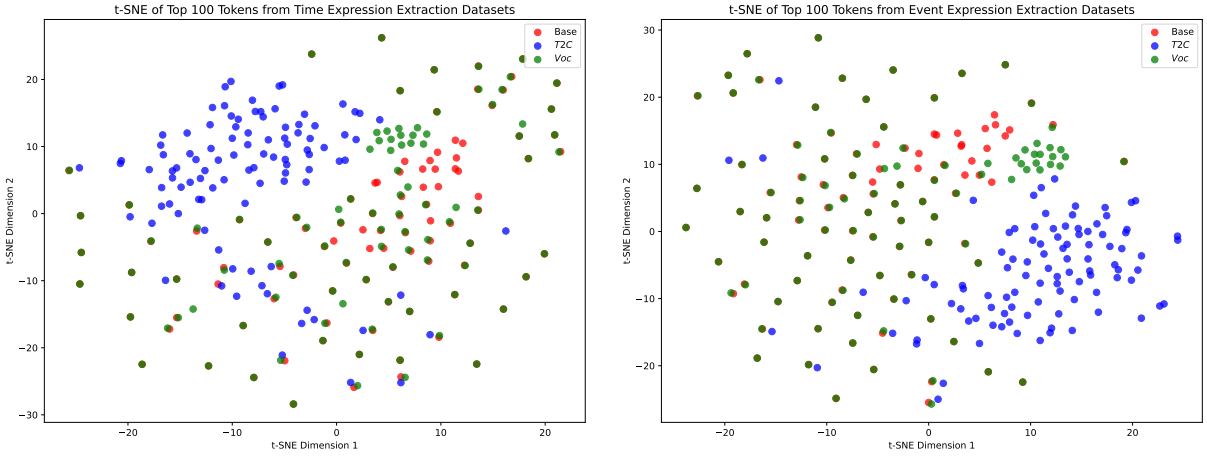


Figure 6: Extraction error analysis on *T2C*, pooled across all reasoning strategies.



(a) TIMEX datasets.

(b) EVENTX datasets.

Figure 7: t-SNE projections of final-layer `roberta-base` token embeddings from the Base, *T2C*, and *Voc* datasets.

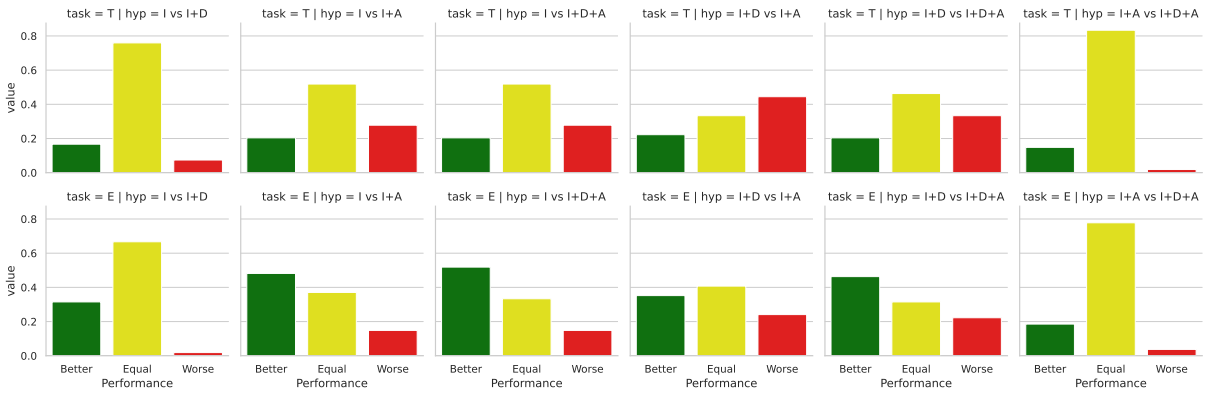


Figure 8: Percentage of models that perform significantly better, worse, or almost equal for reasoning strategy i vs. j (`Hyp = i vs j`), summed over all generalization dimensions. The upper and lower panels show the results for TIMEX (`T`) and EVENTX (`E`) tasks, respectively.

significantly better, worse, or nearly equal to the closest available MoE model from the same family. Exact official size-matched MoE counterparts are not available, so we use the closest comparable variants: Llama 4 Maverick, Mixtral, and Qwen MoE. Although these MoE models have larger active parameter counts, they remain broadly comparable in scale (Llama: 8B vs. 17B, Mistral: 8B vs. 13B, Qwen: 7B vs. 22B). Thus, worse MoE generalization *despite larger active size* suggests limitations beyond parameter count. Figure 9 shows the results.

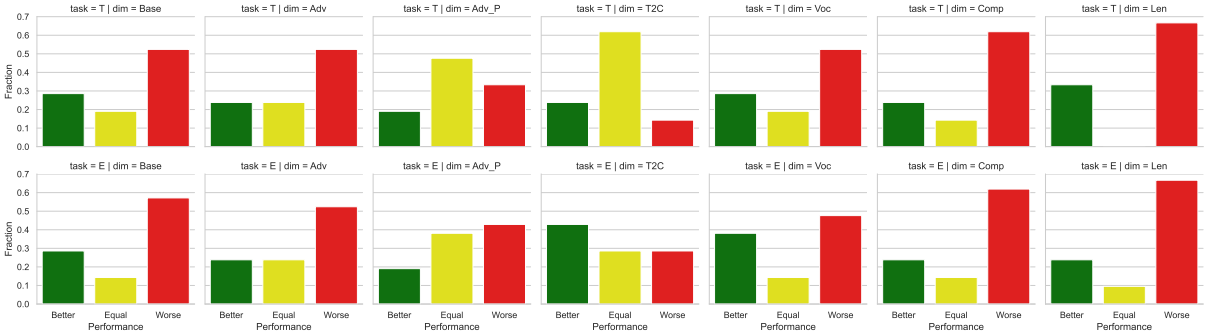


Figure 9: Percentage of Dense models that perform significantly better, worse, or almost equal for the corresponding MoE model in a particular dimension (`dim = x`), summed over all reasoning strategies.

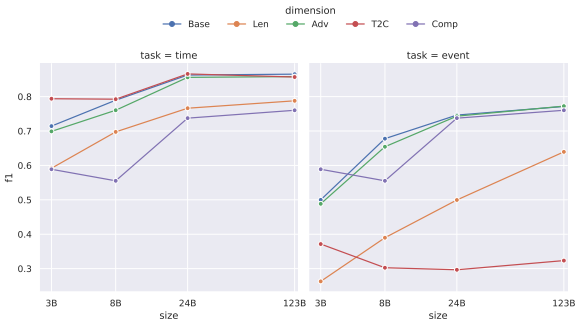


Figure 10: Generalization performance of Mistral Dense models with varying parameter sizes.

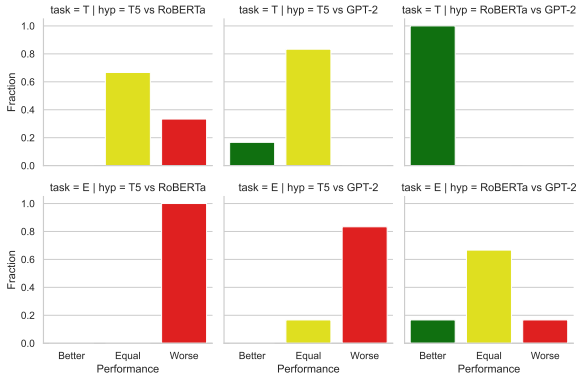


Figure 11: The proportion of TIMEX (upper panel) and EVENTX (lower panel) datasets where a fine-tuned PLM m_i is significantly better, worse, or almost equal to another model m_j (`hyp = `m_i` vs `m_j).

MoE models outperform Dense counterparts on base tasks, but their advantage weakens under larger shifts such as *T2C* and *Adv_P*, where they perform similarly to Dense models, and on EVENTX tasks, where they perform worse. This suggests that higher active parameter counts do not reliably translate into better domain generalization.

Model size. We investigate the effect of model size on generalization with 4 Dense Mistral models of sizes 3B to 123B and a subset of dimensions: a) *Adv*, b) Length, c) *T2C*, and d) Compositional. The results are presented in Figure 10.

Model size correlates well with base-task performance and generalization, but the gains are not uniform across all dimensions. Performance improves monotonically in base, and the same behavior is observed in *Adv* and Length. However, for *T2C* and compositional, the performance difference between smaller changes (3B $\rightarrow$ 8B) is non-significant. Surprisingly, for EVENTX, model size has no effect on domain generalization. In summary, scaling alone does not guarantee consistent gains across all generalization dimensions.

Fine-tuned Models. While our main focus is prompted LLMs, we include a complementary analysis of fine-tuned pretrained models to examine *consistency*, rather than absolute generalization performance. Following Dutt et al. (2024), we train RoBERTa (Liu et al., 2019), T5 (Raffel et al., 2020), and GPT-2 (Radford et al., 2019) on the base TIMEX and EVENTX sequence labeling tasks, and evaluate on one dataset from each domain, length and adversarial dimension. Details are provided in Appendix § A.2. Figure 11 reports the proportion of generalization datasets where model m_i is significantly better, worse, or almost equal to model m_j. Although all three PLMs perform poorly in some generalization dimensions, eg., *Adv_P* and *T2C* (Table 10, §A.5), their relative behavior is stable: RoBERTa consistently outperforms T5 and GPT-2 across all dimensions. This contrasts with prompted LLMs, where no configuration shows comparable cross-dimensional consistency.

4 Related Work

Prior work has studied NLP generalization from multiple perspectives and proposed methods to improve it across dimensions. (Hupkes et al., 2023) provides a broad taxonomy of generalization,

from which we focus on four key categories that cover most benchmark settings. Cross-task generalization captures a model's ability to transfer knowledge across tasks, where instruction-tuned LLMs trained on large-scale annotated data have shown strong performance (Zhang et al., 2026). Models such as GPT-3 (Brown et al., 2020) and LLaMA2 (Touvron et al., 2023) further demonstrate zero-shot capabilities, though concerns about data contamination raise questions about whether these results reflect true generalization (Li and Flanigan, 2024). Beyond task transfer, prior work has also explored reasoning strategies as a lens for evaluating model behavior (Liu et al., 2024). Within the dimensions we study, domain generalization has been widely explored (Ramponi and Plank, 2020; Naik et al., 2022), while robustness is typically evaluated using adversarial perturbations, including both white-box and black-box attacks (Ebrahimi et al., 2018; Jin et al., 2020; Goyal et al., 2023). However, many such methods produce unrealistic inputs and do not capture shortcut learning tendencies (Geirhos et al., 2020). Compositional generalization has been studied across tasks such as machine translation, semantic parsing, and question answering (Dankers et al., 2022; Kim and Linzen, 2020; Keysers et al., 2020), though systematic dataset construction via composition remains limited. Structural generalization, particularly length generalization, has received comparatively less attention, despite consistent evidence that models struggle in "train-short, test-long" settings (Press et al., 2022; Anil et al., 2022; Ruoss et al., 2023).

In parallel, a large body of work has proposed methods to improve generalization. Domain adaptation approaches include gradual fine-tuning across domains and domain adversarial training (Xu et al., 2021; Wright and Augenstein, 2020). Robustness has been addressed through adversarial augmentation, loss regularization, and generative adversarial training (Li et al., 2019; Goodfellow et al., 2015; Kang et al., 2018). For compositionality, earlier work relies on neuro-symbolic methods and meta-learning, while more recent approaches leverage structured prompting in LLMs (Gupta et al., 2020; Lake, 2019; Press et al., 2023). Additionally, several studies explore the integration of inductive, deductive, and abductive reasoning to enhance model capabilities (He et al., 2025; Hu et al., 2025; Cao et al., 2026). In contrast to work that improves generalization methods or evaluates broader temporal reasoning in LLMs (Fatemi et al., 2025; Bazaga et al., 2025; Park et al., 2025), we systematically assess how existing models generalize on the foundational extraction tasks of identifying time and event expressions.

5 Conclusion & Future Work

We present a systematic study of multidimensional generalization in prompted LLMs for time and event expression extraction, covering domain, robustness, compositional, and structural shifts. Our results show that base-task performance is not a reliable proxy for generalization across dimensions. Inductive reasoning strategies improve both base and generalization performance and yield stronger cross-dimensional correlations, but model size and architecture are not consistently predictive of generalization. In contrast, fine-tuned PLMs show more stable relative behavior across dimensions, though not necessarily stronger absolute performance. These findings suggest that reliable temporal extraction requires evaluation beyond in-domain performance and motivate our future work on models and prompting strategies that generalize more consistently across diverse distribution shifts.

Acknowledgments

The authors acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing computational resources that contributed to the research results reported in this paper. More information is available at https://www.tacc.utexas.edu.

Limitations

Our study focuses on extracting time (TIMEX) and event (EVENTX) expressions. These tasks are core primitives for temporal understanding, but they do not directly evaluate higher-level temporal reasoning tasks such as temporal relation inference, timeline construction, or temporal question answering. A second limitation is possible contamination from pretraining corpora, especially for widely used benchmark datasets. However, our main conclusions rely on relative degradation across distribution shifts, where models still show substantial drops, particularly under cross-domain and robustness-oriented settings. Thus, even if some base datasets were partially seen during pretraining, the generalization gaps remain informative.

Our model comparisons also have limits. We refer to all prompted generative models as LLMs for consistency, although some smaller models in our study may also be described as small language models (SLMs). The dense and MoE models differ not only in architecture, but also in active parameter count, pretraining data, and model generation. Therefore, our results should **not** be interpreted as isolating the causal effect of MoE architecture alone. Similarly, our prompting comparison reflects end-to-end prompting strategies as they are commonly used. Because reasoning style, prompt wording, and output format vary together, we cannot fully attribute performance differences to the reasoning strategy alone.

References

Amey Agrawal, Nitin Kedia, Ashish Panwar, Jayashree Mohan, Nipun Kwatra, Bhargav Gulavani, Alexey Tumanov, and Ramachandran Ramjee. 2024. Taming {Throughput-Latency} tradeoff in {LLM} inference with {Sarathi-Serve}. In *18th USENIX symposium on operating systems design and implementation (OSDI 24)*, pages 117–134.

Cem Anil, Yuhuai Wu, Anders Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. 2022. Exploring length generalization in large language models. *Advances in Neural Information Processing Systems*, 35:38546–38556.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Adrián Bazaga, Rexhina Blloshmi, Bill Byrne, and Adrià de Gispert. 2025. Learning to reason over time: Timeline self-reflection for improved temporal reasoning in language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28014–28033.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 995–1005.

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

Mingzi Cao, Xingwei Tan, Mahmud Elahi Akhter, Marco Valentino, Maria Liakata, Xi Wang, and Nikolaos Aletras. 2026. Fundamental reasoning paradigms induce out-of-domain generalization in language models. *arXiv preprint arXiv:2602.08658*.

Taylor Cassidy, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506.

Verna Dankers, Elia Bruni, and Dieuwke Hupkes. 2022. The paradox of the compositionality of natural language: A neural machine translation case study. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4154–4175.

Ritam Dutt, Sagnik Ray Choudhury, Varun Venkat Rao, Carolyn Rose, and VG Vinod Vydiswaran. 2024. Investigating the generalizability of pretrained language models across multiple dimensions: A case study of nli and mrc. In *Proceedings of the 2nd GenBench Workshop on Generalisation (Benchmarking) in NLP*, pages 165–182.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. HotFlip: White-Box Adversarial Examples for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36.

Bahare Fatemi, Seyed Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. 2025. Test of time: A benchmark for evaluating llms on temporal reasoning. In *International Conference on Learning Representations*, volume 2025, pages 94426–94447.

Yao Fu, Leyang Xue, Yeqi Huang, Andrei-Octavian Brabete, Dmitrii Ustiugov, Yuvraj Patel, and Luo Mai. 2024. {ServerlessLLM}:{Low-Latency} serverless inference for large language models. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pages 135–153.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE security and privacy workshops (SPW)*, pages 50–56. IEEE.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M Khapra, and Balaraman Ravindran. 2023. A survey of adversarial defenses and robustness in nlp. *ACM Computing Surveys*, 55(14s):1–39.

Nitish Gupta, Kevin Lin, Dan Roth, Sameer Singh, and Matt Gardner. 2020. Neural module networks for reasoning over text. In *International Conference on Learning Representations*.

Kaiyu He, Mian Zhang, Shuo Yan, Peilin Wu, and Zhiyu Chen. 2025. Idea: Enhancing the rule learning ability of large language model agent through induction, deduction, and abduction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 13563–13597.

Zhiyuan Hu, Yibo Wang, Hanze Dong, Yuhui Xu, Amrita Saha, Caiming Xiong, Bryan Hooi, and Junnan Li. 2025. Beyond'aha!': Toward systematic meta-abilities alignment in large reasoning models. *arXiv preprint arXiv:2505.10554*.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, and 1 others. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, and 1 others. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025.

Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard Hovy. 2018. Adventure: Adversarial training for textual entailment with knowledge-guided examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2418–2428.

Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, and 1 others. 2020. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations*.

Najoung Kim and Tal Linzen. 2020. Cogs: A compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 conference on empirical methods in natural language processing (emnlp)*, pages 9087–9105.

Vasiliki Kougia, Anastasiia Sedova, Andreas Joseph Stephan, Klim Zaporojets, and Benjamin Roth. 2024. Analysing zero-shot temporal relation extraction on clinical notes using temporal consistency. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 72–84.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Brenden M Lake. 2019. Compositional generalization through meta sequence-to-sequence learning. *Advances in neural information processing systems*, 32.

Changmao Li and Jeffrey Flanigan. 2024. Task contamination: Language models may not be few-shot anymore. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18471–18480.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019*. The Internet Society.

Emmy Liu, Graham Neubig, and Jacob Andreas. 2024. An incomplete loop: Instruction inference, instruction following, and in-context learning in language models. *arXiv preprint arXiv:2404.03028*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Aakanksha Naik, Jill Lehman, and Carolyn Rose. 2022. Adapting to the long tail: A meta-analysis of transfer learning research for language understanding tasks. *Transactions of the Association for Computational Linguistics*, 10:956–980.

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.

Yein Park, Chanwoong Yoon, Jungwoo Park, Donghyeon Lee, Minbyul Jeong, and Jaewoo Kang. 2025. Chroknowledge: Unveiling chronological knowledge of language models in multiple domains. In *International Conference on Learning Representations*, volume 2025, pages 5298–5337.

Ofir Press, Noah A Smith, and Mike Lewis. 2022. Train short, test long: Attention with linear biases enables input length extrapolation, 2022. *URL https://arxiv.org/abs/2108.12409*, 5.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711.

James Pustejovsky, Jose Castano, Robert Ingria, Roser Sauri, Robert Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir Radev. 2003. Timebank: A corpus for annotating temporal information. In *Proceedings of the Corpus Linguistics Conference*.

Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo Ponti, and Shay B Cohen. 2024. Are large language model temporally grounded? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7064–7083.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in nlp—a survey. In *Proceedings of the 28th international conference on computational linguistics*, pages 6838–6855.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4902–4912.

Anian Ruoss, Grégoire Delétang, Tim Genewein, Jordi Grau-Moya, Róbert Csordás, Mehdi Bennani, Shane Legg, and Joel Veness. 2023. Randomized positional encodings boost length generalization of transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1889–1903.

Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 142–147.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350.

Walter Simoncini and Gerasimos Spanakis. 2021. Seqattack: On adversarial attacks for named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 308–318.

Statstutor. 2004. Spearman's correlation. Accessed: 2026-03-30.

William F Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C De Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and 1 others. 2014. Temporal annotation in the clinical domain. *Transactions of the association for computational linguistics*, 2:143–154.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Annotating temporal information in clinical narratives. *Journal of biomedical informatics*, 46:S5–S12.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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

Dustin Wright and Isabelle Augenstein. 2020. Transformer based multi-source domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7963–7974.

Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. 2021. Gradual fine-tuning for low-resource domain adaptation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 214–221.

Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. 2023. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, and 1 others. 2026. Instruction tuning for large language models: A survey. *ACM Computing Surveys*, 58(7):1–36.

A Appendix

A.1 Additional Results

A.1.1 Base Task Performance

Figure 12 compares the average Base performance (Micro-F1) of each model under two setups: inductive+ and non-inductive. We observe that inductive+ configurations consistently achieve higher scores across nearly all models in both TIMEX and EVENTX tasks. This suggests that incorporating inductive reasoning, through in-context examples, improves extraction performance on the base dataset.

A.1.2 Correlation Across Generalization Dimensions

Figure 13 presents the configuration-wise rank correlation (ρ) across generalization dimensions for event extraction under inductive+ and non-inductive strategies. In the inductive+ setting (Figure 13a), we observe consistently strong correlations among Base, Adversarial, Vocabulary, and Length dimensions, indicating stable transfer of configuration rankings across these settings. In contrast, the compositional (Comp) and T2C dimensions exhibit notably weaker correlations, suggesting limited alignment with other dimensions. For non-inductive strategies (Figure 13b), while Base, Adversarial, and Vocabulary remain moderately correlated, overall correlations are lower and more variable. Notably, in non-inductive strategies, the T2C dimension shows near-zero or negative correlations with most other dimensions, whereas in inductive+ setups, it shows much higher correlations.

A.1.3 Effect of Reasoning Strategy

Figure 14 shows the percentage of configurations where one reasoning strategy outperforms, underperforms, or remains indistinguishable from another across generalization dimensions for time extraction. Most comparisons fall into the *equal* category, suggesting similar performance across strategies. Inductive-based methods provide small but consistent gains in Base, Adversarial, and Vocabulary settings, while differences become more variable in compositional and cross-domain (T2C) dimensions. Moreover, combining all reasoning types (I+D+A) does not consistently improve performance, indicating diminishing returns from added complexity.

Figure 15 summarizes the same comparison for event extraction. Again, most strategy pairs perform similarly, with inductive variants remaining competitive but less consistently beneficial than in time extraction. Variability increases in compositional and cross-domain settings, where no single strategy consistently dominates.

A.2 Model Training Details

Overview. We fine-tune three pretrained transformer models—RoBERTa, T5, and GPT-2 on two sequence labeling tasks: temporal expression extraction (timex) and event trigger extraction (eventx). Each model is trained independently for each task, resulting in six fine-tuned models. All models are adapted for token classification by adding a linear classification head over token representations. Training is performed using the HuggingFace Transformers library with the Trainer API and the AdamW optimizer.

Task Formulation. Both tasks are formulated as named entity recognition (NER) problems in CoNLL format, where each line contains a token and its corresponding label, and sentences are separated by blank lines. We convert the original annotations into a BIO tagging format. Temporal expressions and event mentions are mapped to token-level labels, where each span is assigned a beginning (B) and inside (I) tag, and all other tokens are labeled as outside (O). Specifically:

- **timex:** B-T, I-T, O
- **eventx:** B-E, I-E, O

Dataset. We train all models on the base temporal expression and event extraction datasets. Each task is represented in CoNLL format with predefined train and test splits. The label encoder is fit on the union of training and test labels to ensure consistent label mappings across splits.

Models. We evaluate three transformer architectures:

- **RoBERTa (roberta-base):** encoder-only model with 12 layers and hidden size 768, implemented using `RobertaForTokenClassification`. Tokenization uses `add_prefix_space=True`.
- **T5 (flan-t5-small):** encoder-decoder model adapted for token classification using `T5ForTokenClassification`, configured to operate as an encoder for this task.

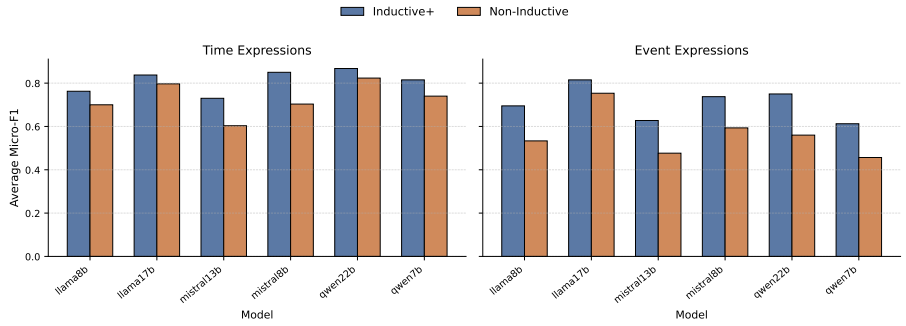
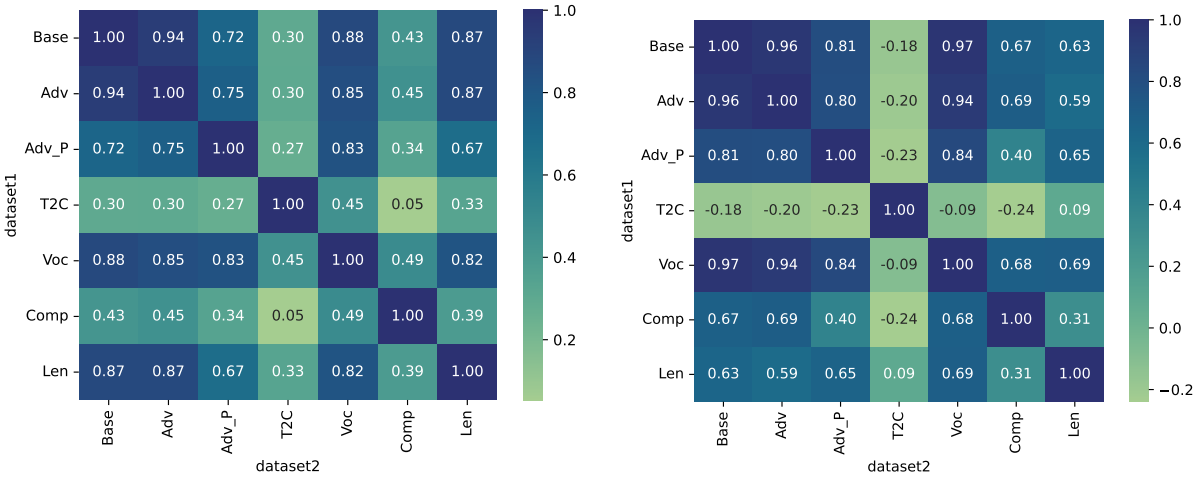


Figure 12: Base task performance in inductive+ vs. non-inductive strategies.



(a) Inductive+ strategies

(b) Non-inductive strategies

Figure 13: ρ across generalization dimensions for event expression extraction in inductive+ and non-inductive strategies.

- **GPT-2 (gpt2):** decoder-only model using `GPT2ForTokenClassification`. The EOS token is used as the padding token, and embeddings are resized using `resize_token_embeddings()`.

Tokenization and Label Alignment. All models use subword tokenization. Word-level labels are aligned to subword tokens by assigning the original label to all subword pieces of a word. Special tokens and padding tokens are assigned an ignore index of -100. All sequences use the maximum context length supported by each model.

Training Configuration. All models are trained with identical hyperparameters:

- Optimizer: AdamW
- Learning rate: 2×10^{-5}
- Weight decay: 0.01
- Warmup steps: 500 (linear schedule with decay)
- Epochs: 3
- Train batch size: 16
- Eval batch size: 32
- Max sequence length: Full model context length

Model checkpoints are saved and evaluated every 250 steps. Early stopping is applied with a patience of 3 evaluations. Model selection is based on entity-level F1 score. All model parameters are updated during training (full fine-tuning); no parameter-efficient fine-tuning methods (e.g., LoRA or adapters) are used.

A.3 Prompting Templates for Reasoning Strategies

In this section, we provide the prompting templates for the three reasoning strategies: inductive (I), deductive (D), and abductive (A)—for the temporal expression extraction task. The same overall prompt structure is used for both temporal and



Figure 14: Percentage of models that perform **significantly** better, worse, or almost equal for reasoning strategy i vs. j, expressed as `Hyp = i vs j`, in each dimension (`Dim = x`) for *time expression extraction tasks*. Each column corresponds to one dimension.

event extraction tasks, although the task definitions and extraction criteria differ for each task. These reasoning prompts are also used consistently across all generalization settings (e.g., domain, compositional, robustness, and structural), ensuring that differences in performance arise from the evaluation conditions rather than changes in prompt design.

A.3.1 Inductive Prompt

Inductive reasoning in our temporal extraction task is formulated as an example-driven span identification process, where the model is provided with annotated instances of valid time expressions, as illustrated in Figure 16 . By observing these examples, the model learns the typical lexical patterns and boundary conventions associated with temporal spans, such as dates, durations, and frequencies. It then generalizes these patterns to extract similar expressions from unseen input text.

A.3.2 Deductive Prompt

Deductive reasoning in our temporal extraction task is formulated as a rule-based span selection process, where the definition of a valid time expression is specified explicitly in the prompt (Figure 17). The model is required to identify spans that directly correspond to grounded temporal references—such as dates, times, durations, and frequencies by strictly applying these criteria to the input text. This setup removes reliance on example-based learning and constrains the model to make extraction decisions based solely on the provided definition, ensuring consistent and precise identification of temporal spans.

A.3.3 Abductive Prompt

In the temporal extraction task, abductive reasoning is implemented as a hypothesis-driven process in which the model generates candidate temporal spans and selects those that best explain the temporal structure of the text (Figure 18). The model is also required to generate explicit explanations describing its selection of the final time expressions. This allows the model to incorporate contextual cues and resolve ambiguities in identifying time expressions. For prompting settings involving abductive reasoning, we additionally provide one structured example to guide smaller models in following the required output format.



Figure 15: Percentage of models that perform **significantly** better, worse, or almost equal for reasoning strategy i vs. j, expressed as `Hyp = i vs j`, in each dimension (`Dim = x`) for *event expression extraction tasks*. Each column corresponds to one dimension.

A.3.4 Inductive + Deductive + Abductive Prompt

Inductive, deductive, and abductive reasoning in our temporal extraction task is formulated as a combined example-guided and hypothesis-driven span identification process (Figure 19). The model first learns from annotated examples which kinds of spans typically function as time expressions and what their boundary conventions look like. It then applies these inferred criteria to the new input text to identify candidate temporal spans. Finally, it selects the spans that best explain the temporal meaning of the text, favoring complete and non-overlapping expressions, while also providing brief reasoning steps for its decisions.

A.4 Datasets

The original document-level TIMEX and EVENTX datasets derived from the TimeBank corpus contain 2.12K training instances, 265 validation instances, and 266 test instances each. From these document-level datasets, we construct sentence-level variants for both tasks. The sentence-level TIMEX dataset contains 22.7K training instances, 2.67K validation instances, and 400 test instances. Similarly, the sentence-level EVENTX dataset contains 23.2K training instances, 3.16K validation instances, and 400 test instances.

The adversarial datasets (*Adv* and *Adv_P*) are generated from the sentence-level test sets for both TIMEX and EVENTX. The vocabulary-shift datasets (*Voc*) are also constructed from the same sentence-level test sets.

For structural generalization (*Len*), we use 80 document-level instances from the original TimeBank datasets for evaluation. These instances have a minimum sequence length of 63 tokens, a maximum sequence length of 1134 tokens, and an average sequence length of 369.98 tokens. Among the 80 instances, 73 contain fewer than 512 tokens, which is the smallest context window among the evaluated PLM models. This helps ensure that our structural generalization results are not affected by the documents being truncated. For all prompted

System Prompt:
You are an expert system for time expression extraction in natural language text.

Task Definition:
You are given annotated examples and a new input text.
Your task is to extract all explicit time expressions from the new input text by generalizing from the examples.
Time expressions may include dates, times, durations, frequencies, or other grounded temporal references explicitly stated in the text.

REASONING TYPE DEFINITION:
You must use inductive reasoning, defined as follows:
Infer which spans qualify as time expressions based on the annotated examples
Learn the typical lexical forms and boundary conventions of these expressions
Apply the inferred patterns consistently to unseen input text
You should not rely on external knowledge or assumptions beyond what can be inferred from the examples.

OUTPUT FORMAT:
Return ONLY the following:
A valid JSON array of strings
Each string is one extracted time expression
No duplicates
Ordered by first appearance in the text
No additional text, explanation, or formatting

Examples:
(Few-shot annotated examples)

Input:
(New context)

Figure 16: Inductive prompting template used for time expression extraction.

System Prompt:
You are an expert system for time expression extraction in natural language text.

Task Definition:
You are given a new input text.
Your task is to extract all explicit time expressions from the text by applying predefined rules. Time expressions may include dates, times, durations, frequencies, or other grounded temporal references explicitly stated in the text.
A span qualifies as a time expression if it explicitly denotes:
A calendar date
A clock time
A duration
A frequency or recurrence
Another grounded temporal reference

REASONING TYPE DEFINITION:
You must use deductive reasoning, defined as follows:
Apply the given definition of time expressions directly to the input text
Select only spans that strictly satisfy the stated criteria
Do not infer new rules or adapt based on the input text
You should rely only on the provided rules and not on examples or learned patterns.

OUTPUT FORMAT:
Return ONLY the following:
A valid JSON array of strings
Each string is one extracted time expression
No duplicates
Ordered by first appearance in the text
No additional text, explanation, or formatting

Input:
(New context)

Figure 17: Deductive prompting template used for time expression extraction.

models, the context length exceeds the maximum sequence length.

For compositional generalization (*Comp*), we select sentence-level instances from the original document-level corpus that contain both time and event expressions. The resulting evaluation set contains 400 instances.

A.4.1 Label and Vocabulary Shift in T2C

For the clinical domain-shift datasets (*T2C*) derived from the 2012 i2b2 Clinical Temporal Relations corpus, the clinical TIMEX dataset contains 1.32K training instances, 1.35K validation instances, and 400 test instances. The clinical EVENTX dataset contains 5.56K training instances, 1.35K validation instances, and 400 test instances.

The statistics in Table 1 highlight substantial distributional differences between the original TimeBank labels (news-oriented corpus) and the clinical *T2C* labels, which reflect the specialized language and domain-specific characteristics of clinical narratives.

A.4.2 Time Expression extraction

We evaluate time expression extraction under six different settings: *Base*, *Adversarial*, *Domain*, *Domain-Vocab*, *Compositional*, and *Structural*. The base setting uses sentence-level examples from TimeBank, while the adversarial setting adds perturbations to test robustness. The domain and domain-vocabulary settings measure generalization to clinical and modified vocabulary data, and the compositional setting requires extracting both time expressions and event triggers together. The structural setting uses longer document-level inputs with multiple temporal references and more complex context. Figure 20 shows example instances from each setting. These settings provide a comprehensive evaluation framework for analyzing temporal extraction generalization across diverse conditions.

Table 1: Top 10 most frequent time and event expressions across the Base (*Base*) and clinical (*T2C*) datasets.

Rank	Base TIMEX	*T2C* TIMEX	Base EVENTX	*T2C* EVENTX
1	friday (25)	the time (9)	said (187)	admission (279)
2	tuesday (25)	the day (9)	talks (27)	discharge (237)
3	monday (24)	that time (5)	told (20)	showed (139)
4	wednesday (17)	the night (5)	take (20)	admitted (122)
5	thursday (17)	day of life two (5)	meeting (18)	transferred (116)
6	now (14)	one (4)	war (14)	revealed (74)
7	sunday (12)	the time of discharge (4)	killed (13)	discharged (59)
8	saturday (12)	daily (4)	expected (13)	presented (53)
9	this year (7)	few days (4)	set (13)	the floor (43)
10	five years (6)	bid (4)	deal (13)	intubated (42)

System Prompt:
You are an expert system for time expression extraction in natural language text.

Task Definition:
You are given a new input text.
Your task is to extract all explicit time expressions by using abductive reasoning. Time expressions may include dates, times, durations, frequencies, or other grounded temporal references explicitly stated in the text.

REASONING TYPE DEFINITION:
You must use abductive reasoning and think step by step, defined as follows:
Generate candidate spans that could plausibly express temporal information
Reason about how each candidate explains the timing, duration, or recurrence of events in the text
Select the spans that best account for the author's intended temporal meaning
Prefer the most complete and non-overlapping spans

OUTPUT FORMAT:
Return ONLY a valid JSON list of dictionaries.
Each dictionary must contain the following fields:
`"text"`: the original input text
`"time_expressions"`: a list of extracted time expression strings (verbatim from the text). If no time expressions are present, `"time_expressions"` must be an empty list ([])
`"reasoning_steps"`: a list of strings explaining the abductive reasoning process

Example Output:

```
[
  {
    "text": "The conference started on April 9, 1997 and lasted for
    three days.",
    "time_expressions": ["April 9, 1997", "three days"],
    "reasoning_steps": [
      "Identified a calendar date.",
      "Identified a duration phrase.",
      "Selected both spans."
    ]
  }
]
```

Input:
(New context)

Figure 18: Abductive prompting template used for time expression extraction.

A.4.3 Event Expression extraction

Unlike time expression extraction, the event extraction task focuses on identifying event trigger words that describe actions, occurrences, or state changes in text. We evaluate the same six settings to analyze how well models generalize under perturbations, domain shifts, vocabulary changes, compositional reasoning, and longer contextual inputs. Figure 21 presents representative examples from each dataset setting, with adversarial perturbations highlighted in red.

A.5 Detailed Results

This section presents the complete experimental results across all extraction tasks, model families, reasoning strategies, and generalization settings. We include detailed results in Tables 2–10 for both time expression extraction and event extraction using dense and MoE models, along with compositional extraction experiments that jointly evaluate temporal and event understanding. The reported scores provide a fine-grained view of model behavior under domain shift, adversarial perturbations, vocabulary variation, and longer contextual inputs. We also analyze the effect of model scale to understand how performance changes with increasing parameter size across different evaluation settings. In addition, we report results for fine-tuned models to compare task-specific adaptation against prompting-based reasoning strategies. These detailed tables complement the main analysis by exposing performance consistency, robustness trends, and setting-specific weaknesses that are less visible from aggregated summaries alone.

Model	Reasoning	Base	*T2C*	*Adv*	Length	*Adv_P*	*Voc*
	D	0.75	0.83	0.78	0.75	0.75	0.73
	I	0.82	0.88	0.89	0.79	0.77	0.78
	A	0.83	0.86	0.84	0.79	0.81	0.84
Llama 17B	D+I	0.82	0.82	0.80	0.69	0.78	0.79
	D+A	0.81	0.82	0.78	0.79	0.83	0.84
	I+A	0.84	0.87	0.89	0.79	0.85	0.89
	D+I+A	0.87	0.80	0.82	0.75	0.83	0.88
	D	0.61	0.76	0.61	0.49	0.64	0.63
	I	0.70	0.76	0.67	0.59	0.74	0.69
	A	0.61	0.78	0.57	0.23	0.68	0.59
Mistral 13B	D+I	0.70	0.75	0.70	0.63	0.70	0.74
	D+A	0.59	0.73	0.60	0.19	0.70	0.58
	I+A	0.78	0.79	0.77	0.48	0.77	0.77
	D+I+A	0.74	0.78	0.74	0.46	0.75	0.70
	D	0.73	0.84	0.79	0.77	0.79	0.80
	I	0.80	0.84	0.88	0.80	0.83	0.88
	A	0.87	0.86	0.87	0.76	0.86	0.87
Qwen 22B	D+I	0.89	0.75	0.77	0.78	0.85	0.87
	D+A	0.87	0.86	0.87	0.77	0.88	0.87
	I+A	0.89	0.87	0.90	0.80	0.89	0.90
	D+I+A	0.89	0.86	0.89	0.78	0.87	0.89

Table 2: Time expression extraction results for MoE models under different reasoning strategies. Scores are reported using micro F1.

Model	Reasoning	Base	*T2C*	*Adv*	Length	*Adv_P*	*Voc*
	D	0.77	0.24	0.77	0.66	0.50	0.71
	I	0.82	0.23	0.82	0.66	0.51	0.70
	A	0.72	0.22	0.75	0.54	0.53	0.74
Llama 17B	D+I	0.80	0.24	0.77	0.60	0.50	0.70
	D+A	0.77	0.23	0.79	0.43	0.54	0.77
	I+A	0.82	0.26	0.83	0.60	0.56	0.81
	D+I+A	0.82	0.25	0.83	0.59	0.56	0.83
	D	0.46	0.45	0.47	0.21	0.35	0.48
	I	0.64	0.26	0.63	0.38	0.49	0.65
	A	0.46	0.28	0.47	0.07	0.44	0.45
Mistral 13B	D+I	0.64	0.27	0.64	0.38	0.49	0.64
	D+A	0.51	0.25	0.50	0.09	0.47	0.50
	I+A	0.63	0.21	0.62	0.19	0.51	0.62
	D+I+A	0.60	0.19	0.62	0.16	0.50	0.61
	D	0.54	0.43	0.54	0.36	0.40	0.58
	I	0.71	0.27	0.70	0.65	0.54	0.80
	A	0.50	0.45	0.50	0.30	0.35	0.51
Qwen 22B	D+I	0.70	0.28	0.75	0.52	0.51	0.68
	D+A	0.64	0.37	0.65	0.15	0.46	0.64
	I+A	0.80	0.28	0.80	0.51	0.55	0.80
	D+I+A	0.79	0.28	0.79	0.53	0.54	0.80

Table 3: Event expression extraction results for MoE models under different reasoning strategies. Scores are reported using micro F1.

Model	Reasoning	Base	*T2C*	*Adv*	Length	*Adv_P*	*Voc*
	D	0.67	0.79	0.66	0.59	0.70	0.67
	I	0.76	0.82	0.75	0.62	0.76	0.76
	A	0.72	0.83	0.71	0.30	0.76	0.72
Llama 8B	D+I	0.76	0.82	0.76	0.62	0.77	0.78
	D+A	0.71	0.83	0.70	0.30	0.77	0.71
	I+A	0.76	0.83	0.76	0.58	0.82	0.77
	D+I+A	0.77	0.84	0.78	0.56	0.83	0.79
	D	0.57	0.74	0.56	0.64	0.65	0.56
	I	0.85	0.80	0.83	0.74	0.77	0.84
	A	0.77	0.82	0.75	0.68	0.77	0.75
Mistral 8B	D+I	0.85	0.80	0.81	0.73	0.82	0.84
	D+A	0.77	0.81	0.73	0.75	0.78	0.76
	I+A	0.84	0.84	0.86	0.78	0.78	0.82
	D+I+A	0.86	0.84	0.86	0.76	0.82	0.85
	D	0.76	0.81	0.75	0.58	0.75	0.76
	I	0.83	0.85	0.81	0.70	0.81	0.84
	A	0.73	0.85	0.74	0.22	0.68	0.54
Qwen 7B	D+I	0.84	0.85	0.81	0.71	0.81	0.85
	D+A	0.73	0.85	0.73	0.21	0.65	0.60
	I+A	0.80	0.83	0.79	0.44	0.67	0.55
	D+I+A	0.79	0.83	0.78	0.37	0.69	0.58

Table 4: Time expression extraction results for dense models under different reasoning strategies. Scores are reported using micro F1.

Model	Reasoning	Base	*T2C*	*Adv*	Length	*Adv_P*	*Voc*
	D	0.55	0.35	0.52	0.35	0.46	0.54
	I	0.74	0.27	0.75	0.48	0.54	0.76
	A	0.54	0.29	0.57	0.33	0.49	0.55
Llama 8B	D+I	0.75	0.28	0.75	0.46	0.55	0.75
	D+A	0.51	0.21	0.48	0.08	0.34	0.50
	I+A	0.63	0.26	0.66	0.33	0.50	0.65
	D+I+A	0.66	0.26	0.68	0.26	0.52	0.64
	D	0.65	0.41	0.66	0.49	0.50	0.64
	I	0.76	0.38	0.76	0.57	0.52	0.78
	A	0.48	0.36	0.52	0.38	0.41	0.48
Mistral 8B	D+I	0.76	0.34	0.76	0.54	0.53	0.76
	D+A	0.65	0.30	0.64	0.05	0.48	0.65
	I+A	0.75	0.30	0.75	0.50	0.52	0.75
	D+I+A	0.68	0.31	0.75	0.43	0.44	0.68
	D	0.53	0.27	0.54	0.23	0.48	0.55
	I	0.68	0.29	0.70	0.29	0.48	0.69
	A	0.35	0.24	0.37	0.06	0.30	0.26
Qwen 7B	D+I	0.65	0.25	0.66	0.24	0.48	0.65
	D+A	0.49	0.22	0.51	0.05	0.29	0.36
	I+A	0.56	0.20	0.60	0.07	0.43	0.42
	D+I+A	0.56	0.19	0.61	0.07	0.41	0.41

Table 5: Event expression extraction results for dense models under different reasoning strategies. Scores are reported using micro F1.

Model	Reasoning	Base	*Adv*	*T2C*	Length
Mistral 3B	D	0.60	0.59	0.44	0.36
	I	0.68	0.64	0.44	0.46
	A	0.28	0.28	0.41	0.13
	D+I	0.67	0.65	0.35	0.41
	D+A	0.31	0.29	0.30	0.05
	I+A	0.39	0.37	0.32	0.22
	D+I+A	0.57	0.60	0.35	0.21
Mistral 24B	D	0.71	0.71	0.32	0.57
	I	0.80	0.80	0.33	0.61
	A	0.65	0.64	0.30	0.45
	D+I	0.79	0.78	0.30	0.54
	D+A	0.69	0.69	0.29	0.22
	I+A	0.79	0.79	0.27	0.55
	D+I+A	0.79	0.79	0.26	0.56
Mistral 123B	D	0.79	0.80	0.32	0.69
	I	0.82	0.82	0.32	0.71
	A	0.56	0.59	0.42	0.43
	D+I	0.82	0.82	0.31	0.70
	D+A	0.76	0.78	0.29	0.56
	I+A	0.82	0.83	0.31	0.70
	D+I+A	0.82	0.83	0.31	0.69

Table 6: Event expression extraction results for model size analysis. Scores are reported using micro F1.

Model	Reasoning	Base	*Adv*	*T2C*	Length
Mistral 3B	D	0.62	0.60	0.77	0.52
	I	0.78	0.77	0.79	0.68
	A	0.69	0.68	0.80	0.49
	D+I	0.77	0.77	0.78	0.67
	D+A	0.68	0.64	0.79	0.50
	I+A	0.73	0.73	0.82	0.63
	D+I+A	0.72	0.71	0.81	0.65
Mistral 24B	D	0.80	0.79	0.84	0.70
	I	0.88	0.87	0.86	0.76
	A	0.85	0.85	0.86	0.77
	D+I	0.88	0.88	0.87	0.78
	D+A	0.84	0.83	0.87	0.77
	I+A	0.89	0.89	0.88	0.79
	D+I+A	0.90	0.89	0.88	0.79
Mistral 123B	D	0.85	0.82	0.86	0.70
	I	0.89	0.86	0.84	0.76
	A	0.84	0.84	0.86	0.77
	D+I	0.88	0.88	0.83	0.78
	D+A	0.82	0.82	0.87	0.77
	I+A	0.89	0.89	0.87	0.79
	D+I+A	0.89	0.89	0.87	0.79

Table 7: Time expression extraction results for model size analysis. Scores are reported using micro F1.

Model	Reasoning	F1
Llama 17B	D	0.71
	I	0.75
	A	0.69
	D+I	0.75
	D+A	0.75
	I+A	0.82
	D+I+A	0.80
Mistral 13B	D	0.45
	I	0.62
	A	0.45
	D+I	0.62
	D+A	0.47
	I+A	0.55
	D+I+A	0.54
Qwen 22B	D	0.56
	I	0.82
	A	0.61
	D+I	0.83
	D+A	0.66
	I+A	0.62
	D+I+A	0.83

Table 8: Compositional extraction results for MoE models (time + event). Scores are reported using micro F1.

Model	Reasoning	F1
Llama 8B	D	0.54
	I	0.75
	A	0.59
	D+I	0.76
	D+A	0.58
	I+A	0.69
	D+I+A	0.72
Mistral 8B	D	0.37
	I	0.39
	A	0.53
	D+I	0.40
	D+A	0.65
	I+A	0.77
	D+I+A	0.77
Qwen 7B	D	0.54
	I	0.71
	A	0.51
	D+I	0.71
	D+A	0.54
	I+A	0.68
	D+I+A	0.69

Table 9: Compositional extraction results for dense models (time + event). Scores are reported using micro F1.

Time Expression Extraction (Fine-tuned Models)						
Model	**Base**	***T2C***	***Adv***	**Length**	***Voc***	***Adv_P***
RoBERTa	0.93	0.69	0.92	0.88	0.93	0.79
T5	0.90	0.55	0.89	0.83	0.90	0.79
GPT-2	0.88	0.48	0.86	0.83	0.88	0.74
Event Expression Extraction (Fine-tuned Models)						
Model	**Base**	***T2C***	***Adv***	**Length**	***Voc***	***Adv_P***
RoBERTa	0.85	0.19	0.82	0.82	0.85	0.50
T5	0.70	0.17	0.68	0.66	0.70	0.39
GPT-2	0.84	0.18	0.78	0.85	0.84	0.50

Table 10: Performance of fine-tuned models on time and event expression extraction across multiple generalization dimensions. Scores are reported using micro F1.

System Prompt:
You are an expert system for time expression extraction in natural language text.

Task Definition:
You are given annotated examples and a new input text. Your task is to extract all explicit time expressions from the new input text. Time expressions may include dates, times, durations, frequencies, or other grounded temporal references explicitly stated in the text.

REASONING TYPE DEFINITION:
You must use inductive, deductive, and abductive reasoning, and think step by step, defined as follows:
Inductively infer from the annotated examples which spans qualify as time expressions, including their typical lexical forms and boundary conventions.
Deductively apply the inferred criteria to the new input text to identify candidate spans that satisfy the definition of a time expression.
Abductively select the spans that best explain the temporal meaning intended by the author, preferring the most complete and non-overlapping spans.
You should not rely on external knowledge or assumptions beyond what can be inferred from the examples and the induced criteria.

OUTPUT FORMAT:
Return ONLY a valid JSON list of dictionaries.
Each dictionary must contain the following fields:
`"text"`: the original input text
`"time_expressions"`: a list of extracted time expression strings (verbatim from the text). If no time expressions are present, `"time_expressions"` must be an empty list (`[]`)
`"reasoning_steps"`: a list of strings explaining the reasoning process

Example Output:

```
[
  {
    "text": "The conference started on April 9, 1997 and lasted for
    three days.",
    "time_expressions": ["April 9, 1997", "three days"],
    "reasoning_steps": [
      "Identified a calendar date.",
      "Identified a duration phrase.",
      "Selected both spans."
    ]
  }
]
```

Input:
(New context)

Figure 19: Inductive, deductive, and abductive prompting template used for time expression extraction.

Base

Text: MHB technicians said they started the surveillance mission Thursday to monitor the body temperature of all Macao residents and visitors.

Extracted span: Thursday

Adversarial

Text: MHB technicians tell they launches the monitor remit Thursday to observer the entity warmth of all Macao reside and traveler.

Extracted span: Thursday

Domain(*T2C*)

Text: Pregnancy was IUI assisted for di-di twin gestation and was relatively uncomplicated until mother presented on 12-15, with spontaneous rupture of membranes.

Extracted span: 12-15

Domain-Vocab

Text: MHB technicians said they started the surveillance mission Thursday to monitor the body temperature of all Myfiu residents and visitors.

Extracted span: Thursday

Compositional

Text: The longtime overheating then led to a blowout and the cold air went into the T101 aniline processing tower, which was run under negative pressure and triggered the blast at about 1:45 p.m., and other blasts happened afterwards.

Extracted time and event spans: 1:45 p.m.; overheating; led; blowout; went; run; triggered; blast; blasts; happened

Structural

Text: XIN_ENG_20061129.0012 WASHINGTON, Nov. 29, 2006 (Xinhua) Clippers, Dunleavy reportedly reach four-year deal. The Los Angeles Clippers and coach Mike Dunleavy have reached a four-year contract extension worth more than 21 million U.S. dollars, according to the Los Angeles Times Tuesday. The team and Dunleavy's agent have worked out a deal that will pay him more than 5.3 million U.S. dollars a season. The Clippers have not made the deal public because bonuses tied to future playoff appearances were still being worked out, the Times said. The Clippers improved dramatically under Dunleavy last season, posting their best win total since the franchise's glory days as the Buffalo Braves in the mid-1970s. The team reached the playoffs for the first time in nine seasons and came within one win of reaching the Western Conference final. The Times reported the offer is close to the 6 million U.S. a year Dunleavy's agent initially proposed and would make him among the highest-paid coaches in the NBA. Dunleavy, the NBA's 1999 coach of the year, currently earns 2.5 million U.S. dollars a season, and his deal expires at the end of this season.

Extracted spans: Nov. 29, 2006; Times Tuesday; last season; mid-1970s; nine seasons; the year; season; the end of this season

Figure 20: Example instances from the six dataset settings used in our time expression extraction experiments: base, adversarial, domain, domain-vocab, compositional, and structural. In the adversarial example, perturbed tokens are highlighted in red.

Base

Text: The typhoon, which hit the eastern region of Bicol before dawn Thursday, packing maximum winds of 150 kilometers (93 miles) per hour, knocked out electrical services, telephone lines and water supply, further hampering rescue and relief efforts.
Extracted spans: hit; packing; knocked; hampering

Adversarial

Text: The typhoon, which hit the eastern region of Bicol before dawn Thursday, packing maimum winds of 150 kilometers (93 miles) per hour, knocked out electrical services, telephone lines and kater supply, further hampering rescue and relief efforts.
Extracted spans: hit; packing; knocked; hampering

Domain(*T2C*)

Text: The patient was reportedly alert at scene, and then became unresponsive in route to the hospital.
Extracted spans: alert; unresponsive; the hospital

Domain-Vocab

Text: The typhoon, which hit the eastern region of Tcofa before dawn Thursday, packing maximum winds of 150 kilometers (93 miles) per hour, knocked out electrical services, telephone lines and water supply, further hampering rescue and relief efforts.
Extracted spans: hit; packing; knocked; hampering

Compositional

Text: The longtime overheating then led to a blowout and the cold air went into the T101 aniline processing tower, which was run under negative pressure and triggered the blast at about 1:45 p.m., and other blasts happened afterwards.
Extracted time and event spans: 1:45 p.m.; overheating; led; blowout; went; run; triggered; blast; blasts; happened

Structural

Text: XIN_ENG_20061129.0012 WASHINGTON, Nov. 29, 2006 (Xinhua) Clippers, Dunleavy reportedly reach four-year deal. The Los Angeles Clippers and coach Mike Dunleavy have reached a four-year contract extension worth more than 21 million U.S. dollars, according to the Los Angeles Times Tuesday. The team and Dunleavy's agent have worked out a deal that will pay him more than 5.3 million U.S. doallars a season. The Clippers have not made the deal public because bonuses tied to future playoff appearances were still being worked out, the Times said. The Clippers improved dramatically under Dunleavy last season, posting their best win total since the franchise's glory days as the Buffalo Braves in the mid-1970s. The team reached the playoffs for the first time in nine seasons and came within one win of reaching the Western Conference final. The Times reported the offer is close to the 6 million U.S. a year Dunleavy's agent initially proposed and would make him among the highest-paid coaches in the NBA. Dunleavy, the NBA's 1999 coach of the year, currently earns 2.5 million U.S. dollars a season, and his deal expires at the end of this season.
Extracted spans: reached; according; worked; deal; pay; made; tied; worked; said; improved; posting; reached; came; reaching; reported; offer; proposed; make; earns; deal; expires

Figure 21: Example instances from the six dataset settings used in our event extraction experiments: base, adversarial, domain, domain-vocab, compositional, and structural. In the adversarial example, perturbed tokens are highlighted in red.