# Beyond Pairwise: Global Zero-shot Temporal Graph Generation

**Alon Eirew**[1] **Kfir Bar**[2] **Ido Dagan**[1]
[1]Computer Science Department, Bar-Ilan University
[2]Efi Arazi School of Computer Science, Reichman University
alon.eirew@gmail.com, kfir.bar@runi.ac.il

## Abstract

Temporal relation extraction (TRE) is a fundamental task in natural language processing (NLP) that involves identifying the temporal relationships between events in a document. Despite the advances in large language models (LLMs), their application to TRE remains limited. Most existing approaches rely on *pairwise* classification, where event pairs are classified in isolation, leading to computational inefficiency and a lack of global consistency in the resulting temporal graph. In this work, we propose a novel *zero-shot* method for TRE that generates a document's complete temporal graph in a single step, followed by temporal constraint optimization to refine predictions and enforce temporal consistency across relations. Additionally, we introduce *OmniTemp*, a new dataset with complete annotations for all pairs of targeted events within a document. Through experiments and analyses, we demonstrate that our method outperforms existing zero-shot approaches and offers a competitive alternative to supervised TRE models.

## 1 Introduction

Temporal relation extraction (TRE) is a foundational task in natural language processing (NLP) that supports applications such as event forecasting (Ma et al., 2023), misinformation detection (Lei and Huang, 2023), and medical treatment timeline construction (Yao et al., 2024).

The TRE task is formulated as follows: given a pair of event mentions, identify the temporal relation between them (e.g., before, after, equal, include, is included, vague). The task has seen significant progress in recent years with the development of supervised models (Tan et al., 2023; Niu et al., 2024). However, these models require large amounts of training data, which is scarce in most domains and languages, and difficult to obtain due to the complexity of manually annotating such relations (Pustejovsky and Stubbs, 2011).

Recent advances in large language models (LLMs) have shown strong capabilities in capturing linguistic patterns (Brown et al., 2020), performing multi-step reasoning (Wei et al., 2022), and applying temporal commonsense knowledge (Jain et al., 2023), positioning them as promising tools to address data scarcity through zero-shot learning (Kojima et al., 2022). However, existing zero-shot LLM-based TRE work has focused on pairwise classification (Yuan et al., 2023; Chan et al., 2024). Pairwise methods face significant computational challenges, particularly in real-world scenarios where the goal is to construct a complete timeline of events from a document. In such cases, all event pairs must be classified, resulting in $O(n^2)$ inference calls for $n$ events. This quadratic complexity becomes impractical when using LLMs due to their high computational cost per query. Moreover, because pairwise approaches consider each event pair in isolation, they fail to capture the global temporal structure of the document, often leading to inconsistent or contradictory temporal graphs (Wang et al., 2020). As a result of these challenges, zero-shot applications of LLMs to TRE have largely been regarded as ineffective (Wei et al., 2024; Niu et al., 2024; Chan et al., 2024; Ning et al., 2024).

In contrast, *global* TRE involves predicting the complete set of temporal relations between all event pairs in a document, resulting in a *temporal graph* that captures the holistic temporal structure. This approach enables models to enforce global consistency and jointly reason about relations, essential for accurate temporal understanding. In doing so, it also provides a more scalable alternative to the computational inefficiencies of pairwise modeling.

A critical obstacle for global TRE research is the lack of datasets with *complete* temporal relation annotations for all event pairs. Manual annotation of full temporal graphs is notoriously challenging and traditionally considered infeasible (Naik et al.,



*Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 31429–31447
November 4-9, 2025 

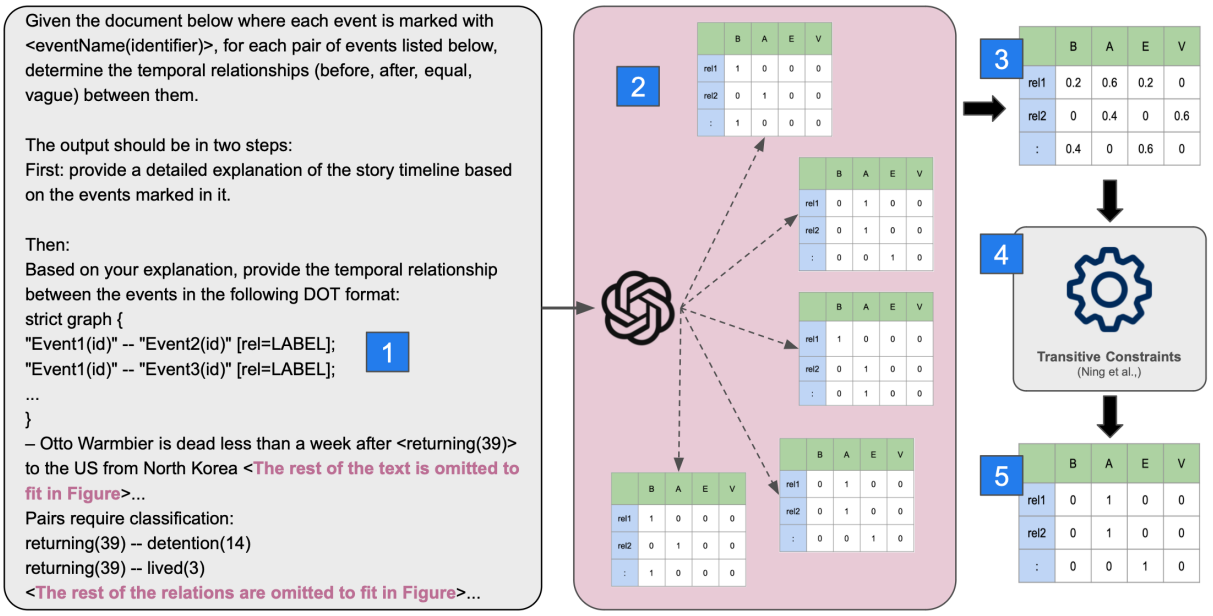



Figure 1: Illustration of the pipeline approach (§4): **[1]** We send the same prompt to the model to generate separate instances of the document's *complete* temporal graph. **[2]** We extract the relation distribution as one-hot vectors over the temporal classes for each relation in each generation. **[3]** We sum and normalize the predictions into a single vector representing the joint prediction over the document's temporal graph. **[4]** We apply an ILP optimization algorithm to this vector. **[5]** The final temporal graph is obtained.

2019). Most TRE datasets therefore provide labels only for a subset of event pairs, often limited to events in consecutive sentences (Chambers et al., 2014; Ning et al., 2018b). This partial coverage constrains models to pairwise strategies and complicates evaluation of long-range temporal reasoning. Alternative automated labeling approaches (Naik et al., 2019; Alsayyahi and Batista-Navarro, 2023) mitigate annotation costs but risk introducing biases inherent to the automated annotation methods themselves.

To address the scarcity of fully annotated datasets, and the inefficiency and global inconsistency of pairwise classification in zero-shot settings, we make the following contributions:[1]

- We propose a novel zero-shot LLM method that generates the entire temporal graph in a single inference step. Our method prompts the model to produce a free-form summary of the event timeline to guide reasoning or "thinking", followed by classification of all event pairs, aggregated via a global temporal constraints optimization algorithm to ensure consistency (Figure 1, and §4). This approach significantly reduces computational cost compared to pairwise methods while achieving performance competitive with supervised models (§6).
- To support global temporal graph extraction, we introduce `OmniTemp`, a new dataset with exhaustive temporal relation annotations for all event pairs (§3).
- We discuss how annotation scope and guideline inconsistencies affect zero-shot model assessment. We show that limiting annotations to short-distance event pairs, as well as discrepancies between widely used datasets such as MATRES and TB-Dense, can hinder fair and reliable evaluation of TRE models in zero-shot settings.

## 2 Background

This section provides relevant background on datasets and zero-shot methods for the temporal relation extraction task.

### 2.1 Temporal Relation Extraction Datasets

The temporal relation extraction task aims to determine the temporal order between pre-extracted events in a text (Pustejovsky et al., 2003). For fair and unbiased model evaluation, datasets should provide gold labels for all event pairs or, at a minimum,

[1]OmniTemp and all experimental code are publicly available at https://github.com/AlonEirew/GlobalZeroShotTRE.

be randomly sampled from the full set. However, most existing datasets for temporal relation extraction provide only partial annotation due to the complexity and cost of the process (Pustejovsky and Stubbs, 2011; Naik et al., 2019). As a result, the two most widely used datasets, MATRES (Ning et al., 2018b) and TimeBank-Dense (TB-Dense) (Chambers et al., 2014), annotate only relations between events in consecutive sentences.

Recently, the NarrativeTime project (Rogers et al., 2024), a large effort of expert annotation, released a comprehensive, re-annotation of the TB-Dense corpus, covering all possible event pairs. The dataset includes seven relation types: *before*, *after*, *includes*, *is-included*, *equal*, *overlap*, and *vague*. Temporal relations are established based on event start times, end times, and durations. Notably, the *vague* relation indicates that the temporal relation cannot be determined from the provided context or where annotators disagree, and it is crucial for *complete* annotation, as it confirms that the pair was considered during annotation and deemed inconclusive, rather than ignored.

While NarrativeTime provides an exhaustively annotated dataset, it follows complex annotation guidelines similar to those of TB-Dense. MATRES refines these guidelines by focusing on a subset of events, annotating relations only based on event start times, and reducing the label set to *before*, *after*, *equal*, and *vague*. These refinements improve inter-annotator agreement and offer a more accessible setting for the task. However, MATRES is not exhaustively annotated. To bridge this gap, we develop OmniTemp, a dataset that adopts the refined MATRES scheme while ensuring complete coverage of *all* event pairs across entire texts. Further details are provided in §3.

### 2.2 Zero-Shot Methods

Recent advancements in LLMs offer an opportunity to leverage their vast knowledge for zero-shot approaches (Kojima et al., 2022), enabling solutions without training data (Zhao et al., 2023). However, few studies have explored LLMs for temporal relation extraction in zero-shot settings. The most notable and best-performing approach is by Yuan et al. (2023), who applied a simple zero-shot chain-of-thought (CoT) method. In this method, the model is sequentially asked about each possible relation for a given event pair (e.g., "Is event-a before event-b?"; if "no," then "Is event-a after event-b?") until the model answers "yes." We use Yuan et al. (2023) method as the zero-shot baseline in our experiments. Another effort by Chan et al. (2024) experimented with prompt engineering and in-context learning. Both methods employed a pairwise approach and achieved suboptimal results on the MATRES and TB-Dense datasets. Additionally, the pairwise approach makes these methods cost- and time-inefficient.

The main goal in this work is to provide a more efficient and effective alternative to pairwise approaches by processing the entire document globally in a single step (see §4).

## 3 The OmniTemp Dataset

OmniTemp[2] is built following the MATRES (Ning et al., 2018b) approach (§2.1); however, instead of annotating events only in consecutive sentences, the annotation is *complete*, covering all event pairs across the entire document. OmniTemp consists of a set of 30 human-generated English news summaries (`Newser.com`), derived from the Multi-News dataset (Fabbri et al., 2019). We select summaries that depict major events (e.g., a presidential visit abroad, a mass shooting, a major earthquake), as these are typically rich in informative sub-event mentions that describe the event timeline. Each summary contains a set of event mentions, with every pair assigned one of the following relations: *before*, *after*, *equal*, or *vague*. We now describe OmniTemp's annotation process (§3.1) along with dataset statistics (§3.2).

### 3.1 Annotation Process

For the annotation process, we hired three non-expert, native English-speaking annotators (students) to label 30 news summaries (~500 words each) for temporal relations (*before*, *after*, *equal*, or *vague*) between salient events, following MATRES guidelines and using the EventFull tool (Eirew et al., 2025).[3] Starting from ~60 auto-detected event mentions per document, extracted using Cattan et al. (2021), annotators selected 15–18 salient events for full-pair annotation, balancing coverage and annotation quality, as prior work shows agreement declines beyond this range for non-expert annotators (Eirew et al., 2025). Final labels were determined by majority vote; in cases of disagree-

[2]Released under a custom license that permits free academic use (see Appendix G).

[3]The complete annotation guidelines are available within the EventFull annotation tool and GitHub repository `https://github.com/AlonEirew/EventFull`.

| | Train | Test | All |
|---|---|---|---|
| Documents | 20 | 10 | 30 |
| Events | 319 | 151 | 470 |
| *before* | 1,119 | 419 | 1,538 |
| *after* | 916 | 431 | 1,347 |
| *equal* | 90 | 60 | 150 |
| *vague* | 276 | 172 | 448 |
| Total Relations | 2,401 | 1,082 | 3,483 |

Table 1: OmniTemp dataset statistics.

ment, the label was set to vague. Further details on the annotation methodology and protocol are provided in Appendix E.

### 3.2 Dataset Statistics and Comparison

Table 1 summarizes the OmniTemp dataset's statistics. Overall, OmniTemp consists of 30 documents, corresponding to 470 event mentions and 3,483 relations. Table 9 in Appendix I, presents the statistics of prominent datasets for the temporal relation extraction task alongside OmniTemp.

The agreement among our annotators averaged 0.72 kappa (Fleiss and Cohen, 1973), corresponding to substantial agreement and is comparable to that of TB-Dense (Chambers et al., 2014) (0.56$\kappa$–0.64$\kappa$), NarrativeTime (Rogers et al., 2024) (0.68$\kappa$), TDD-Manual (Naik et al., 2019) (0.69$\kappa$), and MATRES (Ning et al., 2018b) (0.84$\kappa$). Additionally, to verify annotation accuracy, one of the authors re-annotated 50 random pairs, with 46 matching the majority vote of the annotators, further confirming the high quality of the annotations.

Finally, we assess whether transitivity can compensate for the limited annotation scope in datasets like MATRES and TB-Dense, where only consecutive-sentence pairs are annotated. Using the NarrativeTime dataset, we consider only intra- and consecutive-sentence relations, then apply a transitive closure algorithm (Warshall, 1962) to infer additional links. While some long-distance relations are recovered, most inferred relations remain local and sparse (as illustrated in Figure 11 of Appendix I), further highlighting the importance of exhaustive annotation.

## 4 Zero-Shot Temporal Graph Generation

### 4.1 Prompt Structure

Our zero-shot approach, referred to as *GlobalConsistency*, begins with a straightforward yet powerful idea: prompting an LLM to generate the full temporal graph of a document in a single call (Figure 1). The process starts with a general instruction outlining the task. We then employ a two-step procedure, motivated by the observation that directly prompting the model to classify all event pair relations results in a more inconsistent outputs. To address this, and inspired by reasoning-based prompting techniques (Wang et al., 2023a; Sun et al., 2024), we first prompt the model ([1] in Figure 1) to construct a free-form timeline that summarizes the temporal flow of the marked events. This primes the model with a broader understanding of event order before making explicit classification decisions. We then instruct the model to predict temporal relations between all event pairs. The input includes the full document with event mentions highlighted using angle brackets and unique identifiers (e.g., `<attack(7)>`), followed by a list of all possible event pairs.

For the output, we instruct the model to represent relations as a graph, where events serve as nodes and relations as edges, formatted in the DOT language (Gansner, 2006), which helps suppress free-text explanations and facilitates parsing (an example of the generated timeline is presented in Appendix I, Figure 10).

In documents containing many events that may exceed the model's input capacity, we generate the complete set of pairs and split them evenly for separate processing. Each split receives the same instructions and the full report with all event mentions marked in it, this is followed by only the relevant subset of event pairs for that split. The predictions are then merged back in post-processing (Further details are provided in Appendix A)

### 4.2 Post Process

LLMs are inherently stochastic and may produce different labels for the same input when run multiple times, leading to unstable outputs, especially for ambiguous event pairs. To address this, inspired by self-consistency methods (Wang et al., 2023b) and temporal graph consistency optimization techniques (Ning et al., 2018a), we run the model $M = 5$ times per document, as experimental results show that performance saturates after five generations (see Figure 5 in Appendix A), and aggregate the predicted relation labels into a distribution $p_{ij} \in \mathbb{R}^{|\mathcal{R}|}$ for each event pair $(e_i, e_j) \in \mathcal{E} \times \mathcal{E}$, where $\mathcal{E}$ denotes the set of events, and $p_{ij}^r$ represents the empirical likelihood of label $r \in \mathcal{R}$ across runs. ([2,3] in Figure 1).

We then apply the Integer Linear Programming (ILP) formulation of Ning et al. (2018a) to obtain a globally consistent graph. Specifically, we define binary variables $\mathcal{I}_r(i,j) \in \{0,1\}$ for each relation $r$ and pair $(e_i, e_j)$, and optimize for enforcing key structural constraints: uniqueness (only one relation per pair), symmetry (e.g., if $r = \texttt{BEFORE}$, then its inverse holds for the reverse pair), and transitivity (e.g., if $A \rightarrow B \rightarrow C$, then $A \rightarrow C$) ([4] in Figure 1). The result is the optimal temporal graph that maximizes model confidence while ensuring global coherence. Further details are provided in Appendix F.

## 5 Experimental Setting

We describe the datasets and models used in our experiments. Technical details are in Appendix A.

### 5.1 Datasets

In our experiments, we use our own OmniTemp and three additional datasets: MATRES, TB-Dense, and NarrativeTime.[4] Notably, TCR (Ning et al., 2018a) and TDD-Manual (Naik et al., 2019), two additional datasets for the TRE task, are excluded from our experiments as they omit the *vague* relation. Since we generate relations for all possible event pairs, the *vague* label is essential to avoid forcing incorrect relations when context is insufficient. Below, we provide details on the datasets used in our experiments. For our own OmniTemp, we use the first 10 documents as the test set and the remaining documents as the training set, while for all other datasets, we follow their predefined splits.

**MATRES.** In MATRES, only events within consecutive sentences are annotated. The dataset includes four relation types: *before*, *after*, *equal*, and *vague*, with temporal relations determined based on event start times.

**TB-Dense.** Similar to MATRES, only events within consecutive sentences are annotated in the TB-Dense dataset. It includes six relation types, the four from MATRES plus *includes* and *is-included*. Temporal relations are determined based on event start and end times as well as their duration.

**NT-6.** The NarrativeTime (NT) dataset, previously introduced in §2.1, features seven relation types, including the six from TB-Dense and the *overlap* relation. However, we exclude the *overlap* relation as it is incompatible for the temporal consistency methods, given that the symmetric counterpart was not annotated. Additionally, NT documents contain an average of 50 events, corresponding to 1,200 relations, per document. Due to LLM context limits, we randomly select 18 events per NT document.

### 5.2 Baseline and State-of-the-Art Models

We compare our GlobalConsistency method with four models, reproducing state-of-the-art (SOTA) supervised models and a zero-shot chain-of-thought (CoT) baseline method.

**Bayesian (Tan et al., 2023).** Bayesian-Translation is the current publicly available state-of-the-art pairwise model for temporal relation extraction. It leverages a COMET-BART encoder (Hwang et al., 2020) and a graph translation model (Balazevic et al., 2019) to incorporate prior knowledge from the ATOMIC commonsense knowledge base, refining event representations for relational embedding learning. Additionally, it employs a Bayesian framework to estimate the uncertainty of the learned relations.

**RoBERTa (Tan et al., 2023).** A strong pairwise model for temporal relation extraction, similar in architecture to the Bayesian model described above, but replacing the COMET-BART encoder with a RoBERTa-large encoder (Zhuang et al., 2021). We use this model as it represents a strong, purely supervised approach, allowing for a direct comparison without the influence of external knowledge.

**Bayesian + Constraints.** We extend the Bayesian model with the temporal constraints optimization algorithm (Ning et al., 2018a), the same algorithm used in our GlobalConsistency method, applying it at inference time to enable a more direct comparison with our methods.

**CoT (Yuan et al., 2023).** As a baseline model, we re-implemented the CoT model (Yuan et al., 2023) using GPT-4o and DeepSeek-R1, replacing the original implementation, which used ChatGPT. To the best of our knowledge, this is the strongest zero-shot approach for temporal relation extraction. Unlike our method, which generates relations for all event pairs, the CoT baseline is applied only to event pairs with gold annotations, due to its high computational cost.

For evaluation, we report the F1 score on all datasets following the definition in (Ning et al., 2019), where the *vague* relation is excluded from true positive predictions. Additionally, we report a Temporal Inconsistency (TI) measure by applying a transitive closure algorithm (Warshall, 1962) and

[4] Dataset license details are in Appendix G.

| | *Non-Exhaustive* | | | | *Exhaustive* | | | |
|---|---|---|---|---|---|---|---|---|
| | **MATRES** | | **TB-Dense** | | **NT-6** | | **OmniTemp** | |
| | F1 | TI | F1 | TI | F1 | TI | F1 | TI |
| ***Supervised* SOTA Pairwise Models** | | | | | | | | |
| ∗ RoBERTa (Tan et al.) | 80.4 | 24 | 60.5 | 107 | 59.3 | 105 | 73.6 | 143 |
| ∗ Bayesian (Tan et al.) | **82.7** | 16 | **65.0** | 87 | 64.9 | 203 | 78.7 | 166 |
| + Constraints | – | – | – | – | **65.6** | 0 | **80.7** | 0 |
| ***Zero-Shot* Prompting with *GPT-4o*** | | | | | | | | |
| CoT (Yuan et al.) | 56.6 | – | **42.8** | – | 49.3 | 461 | 67.2 | 374 |
| ZSL-Global (Ours) | 59.0 | 73 | 37.7 | 250 | 48.4 | 300 | 62.3 | 161 |
| ZSL-Timeline (Ours) | 58.4 | 81 | 39.1 | 225 | 52.2 | 309 | 68.5 | 157 |
| SelfConsistency (Ours) | 60.1 | 50 | 41.2 | 122 | 55.6 | 305 | 71.0 | 128 |
| GlobalConsistency (Ours) | **63.0** | 0 | **42.8** | 0 | **58.4** | 0 | **73.6** | 0 |
| ***Zero-Shot* Prompting with *DeepSeek-R1*** | | | | | | | | |
| CoT (Yuan et al.) | **70.3** | – | **50.8** | – | 57.9 | 360 | 78.4 | 254 |
| ZSL-Global (Ours) | 61.0 | 82 | 44.6 | 276 | 57.0 | 262 | 70.5 | 167 |
| ZSL-Timeline (Ours) | 59.0 | 82 | 44.4 | 261 | 59.4 | 185 | 74.6 | 90 |
| SelfConsistency (Ours) | 61.2 | 58 | 46.8 | 152 | 62.1 | 144 | 78.7 | 79 |
| GlobalConsistency (Ours) | 66.4 | 0 | 49.0 | 0 | **64.1** | 0 | **79.2** | 0 |

Table 2: F1 and Transitive Inconsistency (TI) scores of all models on four datasets, grouped into *Non-Exhaustive* Annotation (MATRES, TB-Dense) and *Exhaustive* Annotation (NT-6, OmniTemp). We use the F1 definition from Ning et al. (2019), and compute the average number of TI edges per test document by applying a transitive closure algorithm (Warshall, 1962) and counting transitive contradictions. (∗) For MATRES and TB-Dense with supervised models, we report results from Tan et al. (2023) as our reproductions were slightly lower; TI is based on our retrained models. Constraints are not reported for these models as they did not improve results. (–) TI is not reported for CoT on MATRES and TB-Dense, as it only predicts gold-labeled pairs and cannot construct a complete graph. Computing TI for CoT would require multiple generations over the full set of pairs, which is prohibitively expensive (see Table 3). Further details are provided in Appendix C.

counting transitive contradictions (further details in Appendix C.2).

### 5.3 Ablation Study Design

To investigate the contribution of each component in our method to the overall performance, we design the following three ablation models:

**ZSL-Global.** This zero-shot learning (ZSL) configuration prompts the LLM once to generate the entire temporal graph directly, ommiting the instruction to generate the timeline of events before classification.

**ZSL-Timeline.** This ZSL configuration includes only the prompting step (with timeline generation) but omits the post-processing step that enforces global consistency.

**SelfConsistency.** This configuration replaces our global consistency optimization with a simpler self-consistency approach (Wang et al., 2023b), where the final label for each event pair is selected by majority vote from the five generated outputs.

## 6 Results

Our results are presented in Table 2, with *supervised* SOTA pairwise models shown in the upper section, and the results of our *zero-shot* methods, using GPT-4o and DeepSeek-R1, shown in the lower section.[5] Overall, using GPT-4o, our GlobalConsistency approach (§4) outperforms the CoT baseline (Yuan et al., 2023) by a large margin across all datasets except TB-Dense. Using DeepSeek-R1, our GlobalConsistency method outperforms the CoT baseline on the densely annotated datasets, NT-6 and OmniTemp, but shows lower performance on the sparsely annotated datasets, MATRES and TB-Dense (further analyzed in §7). However, the improved performance of the CoT method comes at a significant cost, ~7× more expensive (Table 3), and requires more time, particularly in the DeepSeek-R1 experiments (Table 4).

[5]Details on additional models and datasets evaluated are provided in Appendix B.

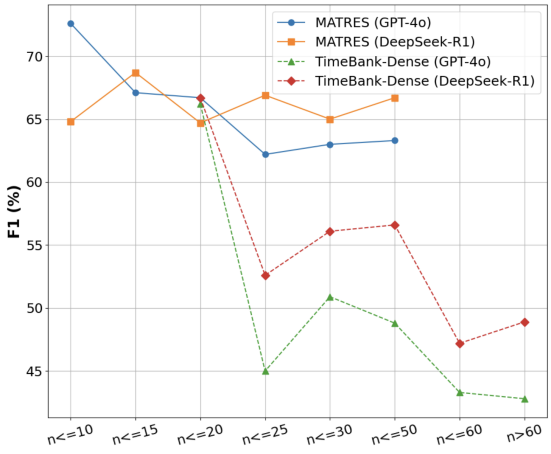



Figure 2: Impact of event count per document on GlobalConsistency performance, evaluated on MATRES and TB-Dense. The x-axis is cumulative, and the y-axis shows the F1 score per subset.

Furthermore, time and cost differences between CoT and our method do not scale linearly with graph size. This is evident in Table 3, where NT-6, with only two more events per document than OmniTemp, incurs higher costs with CoT.

Notably, on the dense datasets, NT-6 and OmniTemp, GlobalConsistency using DeepSeek-R1 matches supervised models (79.2 vs. 80.7 for OmniTemp, 64.1 vs. 65.6 for NT-6), while producing more consistent graphs with lower transitive inconsistency (TI) scores reported in the table. Moreover, our approach requires no training data and does not rely on a substantial external commonsense knowledge base (as required by the Bayesian-Trans model for example), which may not be applicable across many domains and languages. This positions GlobalConsistency as an appealing zero-shot alternative for TRE in scenarios where labeled training data or comprehensive knowledge resources are rare or unavailable.

## 7 Discussion

**Event Mentions Count.** We investigate how the number of events in a document impacts the performance of our GlobalConsistency method. Our hypothesis is that models encoding global information are more sensitive to event count, as they must process more information simultaneously. In contrast, pairwise methods, which consider one event pair at a time, are likely less affected. Figure 2 shows MATRES and TB-Dense documents grouped by increasing event counts.[6] With the exception of DeepSeek-R1 on MATRES, which demonstrates resilience to large event counts, the results show a performance decline as the number of events increases. This supports our hypothesis and may help explain the performance gap between the CoT method and the ZSL-Global variant in most tests.[7]

| | CoT | | GlobalConsistency | |
|---|---|---|---|---|
| | GPT-4o | DeepSeek-R1 | GPT-4o | DeepSeek-R1 |
| MATRES | 50 | 69 | 6 | 9 |
| TB-Dense | 71 | 99 | 9 | 17 |
| NT-6 | 15 | 21 | 2 | 3 |
| OmniTemp | 12 | 16 | 2 | 3 |

Table 3: Approximate costs (USD) for the full dataset are shown. For GlobalConsistency, the cost reflects five generations of the complete set of relations. For the CoT method, to reflect a real-world scenario, we generate the complete set of relations once—rather than just the gold ones. Costs are computed using token counts (via OpenAI's `tiktoken`, and DeepSeek official tokenizer) and official model pricing.

| | CoT | | GlobalConsistency | |
|---|---|---|---|---|
| | GPT-4o | DeepSeek-R1 | GPT-4o | DeepSeek-R1 |
| MATRES | 297 | 2,789 | 250 | 363 |
| TB-Dense | 426 | 4,004 | 360 | 521 |
| NT-6 | 89 | 838 | 75 | 110 |
| OmniTemp | 70 | 658 | 60 | 86 |

Table 4: Time (in minutes) to generate the full set of temporal relations for each test set. For GlobalConsistency, this includes five generations; for CoT, it reflects a single pass over all relations (not just gold) to simulate real-world use.

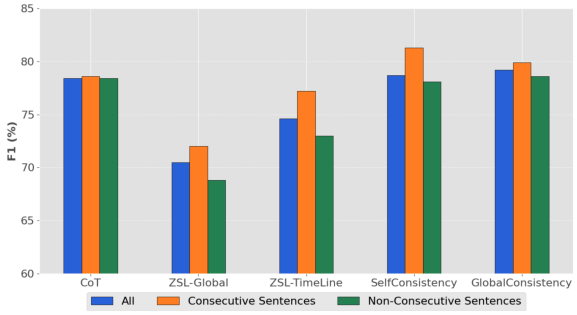
**Event Pair Distance.** We examine whether the annotation distance restriction, where events are annotated only if they are at most one sentence apart, as in MATRES and TB-Dense, can affect model evaluation. To explore this, we evaluate all zero-shot methods on three subsets of OmniTemp and NT-6: the full dataset, event pairs with a sentence distance of at most one (consecutive sentences like in MATRES and TB-Dense), and event pairs with a sentence distance greater than one (non-consecutive sentences). See Figure 3.

Our findings show that on the four-relation OmniTemp dataset, the CoT baseline performs consistently across all sentence distances, while our

[6] The other datasets we experimented with contain a limited number of events per instance.

[7] In TB-Dense, performance drops sharply for documents with over 25 events. For further analysis, see Appendix C.3.

(a) OmniTemp.




(b) NarrativeTime.

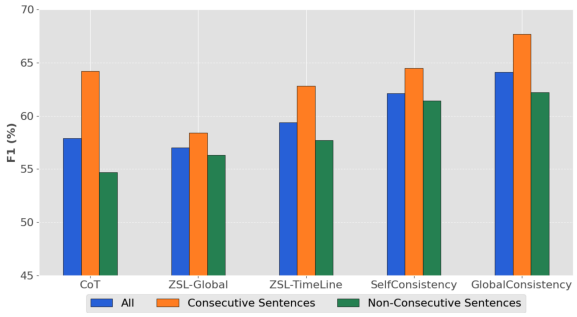
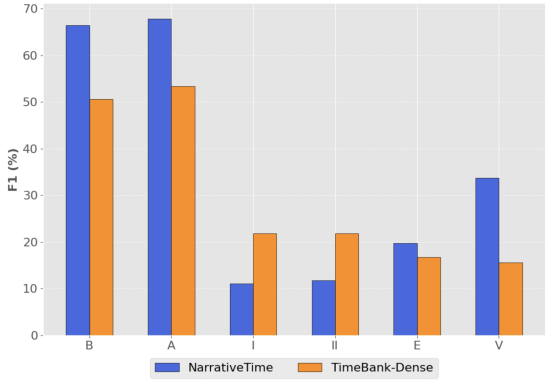



Figure 3: DeepSeek-R1 model performance across different relation subsets: (1) consecutive-sentence event pairs, (2) non-consecutive-sentence event pairs, and (3) full-document event relations. A similar figure for GPT-4o is presented in Figure 6 in Appendix I.

(a) NarrativeTime Vs. TimeBank-Dense.




(b) OmniTemp Vs. MATRES.

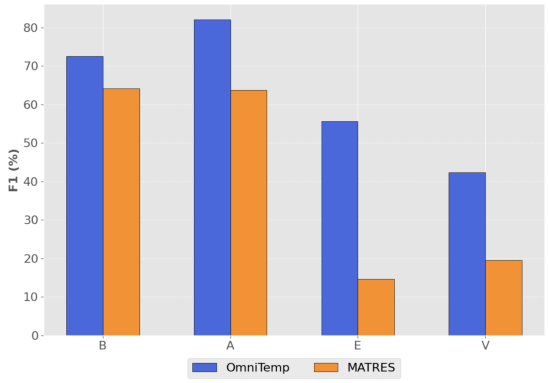



Figure 4: We examine the performance of our prompting method (i.e., ZSL-Timeline) by relation type across two groups of datasets with similar annotation schemes: six-label datasets (TB-Dense and NT-6) and four-label datasets (MATRES and OmniTemp), using DeepSeek-R1. Similar results are observed with GPT-4o, as presented in Figure 7 in Appendix I. The relation labels are: A = *after*, B = *before*, I = *includes*, II = *is-included*, E = *equal*, and V = *vague*.

global methods achieve higher performance on consecutive-sentence pairs. In contrast, on the more challenging six-relation NT-6 dataset, CoT performs notably better on consecutive-sentence pairs than on long-distance pairs. These findings highlights the importance of document-level annotations for reliable evaluation of temporal relation classification—especially in zero-shot settings, where models cannot realistically rely on distribution patterns in the annotations.

**Label Inconsistency.** The performance gap between our methods and the supervised models varies across datasets, being more pronounced in MATRES and TB-Dense than in NT-6 and OmniTemp. To better understand this gap, we analyze the ZSL-Timeline variant (chosen to isolate the model's performance without the influence of post-processing) by examining results per label and grouping datasets with similar label sets, as shown in Figure 4. Our ZSL-Timeline method performs significantly worse on MATRES and TB-Dense than on OmniTemp and NT-6.

To investigate this further, we examine label consistency in documents and event pairs shared between TB-Dense and MATRES, which annotated the same corpus. There are 983 such event pairs. While these datasets follow different annotation guidelines, certain labels should remain consistent. For instance, if an event pair is labeled *equal* in TB-Dense, indicating that both the start and end times of the two events are the same, then the relation should also be *equal* in MATRES. Measuring consistency across the four shared relations, we find strong agreement for *before* and *after*, with *before* being the most consistently annotated. However, significant inconsistencies were evident in *vague* and *equal* (detailed results are provided in Appendix D). Since in zero-shot settings the model is not trained on a dataset, it does not learn dataset-specific annotation biases. The annotation inconsistency between MATRES and TB-Dense may partly explain the performance drop on these datasets, particularly for *vague* and *equal* relations, as well as

the lower performance on *after* compared to *before*.

This analysis, together with the pair distance analysis, may help explain the gap observed between the zero-shot and supervised methods on MATRES and TB-Dense, raising a broader question about the reliability of evaluating zero-shot approaches on partially annotated or inconsistent resources.

## 8 Conclusion

In this work, we introduced a novel zero-shot LLM approach for temporal relation extraction that generates the entire temporal graph at once. Our method moves beyond traditional pairwise approaches, which suffer from computational inefficiency and lack of global consistency. To ensure temporal consistency, we incorporated temporal constraints optimization, significantly improving both accuracy and efficiency while generating relations completely free of inconsistencies. Our results show that zero-shot LLMs, when prompted to generate the timeline of events in free-form language before assigning labels to event pairs and extended with a global constraints algorithm, can serve as a competitive alternative to supervised models, especially in low-resource or cross-domain settings where training data is scarce. Additionally, we introduced *OmniTemp*, a new dataset with complete annotations for all event pairs, following the refined annotation guidelines of MATRES. By providing gold labels for every event pair in a document, this dataset enables a fair evaluation of zero-shot approaches.

## Limitations

While our proposed zero-shot temporal graph generation approach demonstrates significant advantages over pairwise methods, several limitations remain that warrant further investigation.

First, closed LLMs such as GPT-4o and DeepSeek-R1 do not disclose their training data. Therefore, results on the three datasets we investigate may be affected by potential data contamination if their test sets were included in the training phase. However, OmniTemp is a completely new resource that is not yet publicly available, ensuring uncontaminated results.

Second, although self-consistency prompting mitigates stochasticity to some extent, the model's responses can still be inconsistent, especially when handling long-distance temporal dependencies or ambiguous event relations.

Finally, the computational cost of using LLMs for large-scale inference remains a challenge. While our approach significantly reduces costs compared to pairwise methods, generating a full temporal graph for documents with many events can still be time-intensive and expensive.

Despite these limitations, our study highlights promising directions for leveraging LLMs in structured event reasoning and lays the groundwork for future improvements in temporal relation extraction.

## Acknowledgments

This work was supported by the Israel Science Foundation (grant no. 2827/21), and by funding from the Israeli Planning and Budgeting Committee (PBC).

We also used AI-based assistance (ChatGPT) for grammar and spelling corrections during manuscript preparation.

## References

James F. Allen. 1984. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154.

Sarah Alsayyahi and Riza Batista-Navarro. 2023. TIMELINE: Exhaustive annotation of temporal relations supporting the automatic ordering of events in news articles. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16336–16348, Singapore. Association for Computational Linguistics.

Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. Multi-relational poincaré graph embeddings. In *Neural Information Processing Systems*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Arie Cattan, Alon Eirew, Gabriel Stanovsky, Mandar Joshi, and Ido Dagan. 2021. Cross-document coreference resolution over predicted mentions. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5100–5107, Online. Association for Computational Linguistics.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284.

Chunkit Chan, Cheng Jiayang, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2024. Exploring the potential of ChatGPT on sentence level relations: A focus on temporal, causal, and discourse relations. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 684–721, St. Julian's, Malta. Association for Computational Linguistics.

Alon Eirew, Eviatar Nachshoni, Aviv Slobodkin, and Ido Dagan. 2025. EventFull: Complete and consistent event relation annotation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 494–508, Albuquerque, New Mexico. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Joseph L. Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619.

Emden R. Gansner. 2006. Drawing graphs with dot.

Gurobi Optimization, LLC. 2024. Gurobi Optimizer Reference Manual.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2020. COMET-ATOMIC 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI Conference on Artificial Intelligence*.

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, Singapore. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Yuanyuan Lei and Ruihong Huang. 2023. Identifying conspiracy theories news based on event relation graph. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9811–9822, Singapore. Association for Computational Linguistics.

Yunshan Ma, Chenchen Ye, Zijian Wu, Xiang Wang, Yixin Cao, and Tat seng Chua. 2023. Context-aware event forecasting via graph disentanglement. *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Aakanksha Naik, Luke Breitfeller, and Carolyn Rose. 2019. TDDiscourse: A dataset for discourse-level temporal ordering of events. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 239–249, Stockholm, Sweden. Association for Computational Linguistics.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018a. Joint reasoning for temporal and causal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2278–2288, Melbourne, Australia. Association for Computational Linguistics.

Qiang Ning, Sanjay Subramanian, and Dan Roth. 2019. An improved neural baseline for temporal relation extraction. 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 6203–6209, Hong Kong, China. Association for Computational Linguistics.

Qiang Ning, Hao Wu, and Dan Roth. 2018b. 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, Melbourne, Australia. Association for Computational Linguistics.

Wanting Ning, Lishuang Li, Xueyang Qin, Yubo Feng, and Jingyao Tang. 2024. Temporal cognitive tree: A hierarchical modeling approach for event temporal relation extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 855–864, Miami, Florida, USA. Association for Computational Linguistics.

Jingcheng Niu, Saifei Liao, Victoria Ng, Simon De Montigny, and Gerald Penn. 2024. ConTempo: A unified temporally contrastive framework for temporal relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1521–1533, Bangkok, Thailand. Association for Computational Linguistics.

James Pustejovsky, José M Castano, Robert Ingria, Roser Sauri, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R Radev. 2003. TimeML: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

James Pustejovsky and Amber Stubbs. 2011. Increasing informativeness in temporal annotation. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 152–160, Portland, Oregon, USA. Association for Computational Linguistics.

Anna Rogers, Marzena Karpinska, Ankita Gupta, Vladislav Lialin, Gregory Smelkov, and Anna Rumshisky. 2024. NarrativeTime: Dense temporal annotation on a timeline. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12053–12073, Torino, Italia. ELRA and ICCL.

Simeng Sun, Yang Liu, Shuohang Wang, Dan Iter, Chenguang Zhu, and Mohit Iyyer. 2024. PEARL: Prompting large language models to plan and execute actions over long documents. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 469–486, St. Julian's, Malta. Association for Computational Linguistics.

Xingwei Tan, Gabriele Pergola, and Yulan He. 2023. Event temporal relation extraction with Bayesian translational model. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1125–1138, Dubrovnik, Croatia. Association for Computational Linguistics.

Xingwei Tan, Yuxiang Zhou, Gabriele Pergola, and Yulan He. 2024. Set-aligning framework for auto-regressive event temporal graph generation. 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 3872–3892, Mexico City, Mexico. Association for Computational Linguistics.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020. Joint constrained learning for event-event relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 696–706, Online. Association for Computational Linguistics.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023a. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2609–2634, Toronto, Canada. Association for Computational Linguistics.

Xiaozhi Wang, Yulin Chen, Ning Ding, Hao Peng, Zimu Wang, Yankai Lin, Xu Han, Lei Hou, Juanzi Li, Zhiyuan Liu, Peng Li, and Jie Zhou. 2022. MAVEN-ERE: A unified large-scale dataset for event coreference, temporal, causal, and subevent relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 926–941, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Stephen Warshall. 1962. A theorem on boolean matrices. *J. ACM*, 9(1):11–12.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Kangda Wei, Aayush Gautam, and Ruihong Huang. 2024. Are LLMs good annotators for discourse-level event relation extraction? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1–19, Miami, Florida, USA. Association for Computational Linguistics.

Jiarui Yao, Harry Hochheiser, WonJin Yoon, Eli Goldner, and Guergana Savova. 2024. Overview of the 2024 shared task on chemotherapy treatment timeline extraction. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, pages 557–569, Mexico City, Mexico. Association for Computational Linguistics.

Chenhan Yuan, Qianqian Xie, and Sophia Ananiadou. 2023. Zero-shot temporal relation extraction with ChatGPT. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 92–102, Toronto, Canada. Association for Computational Linguistics.

Xuandong Zhao, Siqi Ouyang, Zhiguo Yu, Ming Wu, and Lei Li. 2023. Pre-trained language models can be fully zero-shot learners. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15590–15606, Toronto, Canada. Association for Computational Linguistics.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

## A Experimental Details

For all supervised model experiments, we follow the experimental setup of Tan et al. (2023). To this end, we conducted a grid search to determine the optimal hyperparameters and embedding dimensionality for each test. Each training episode

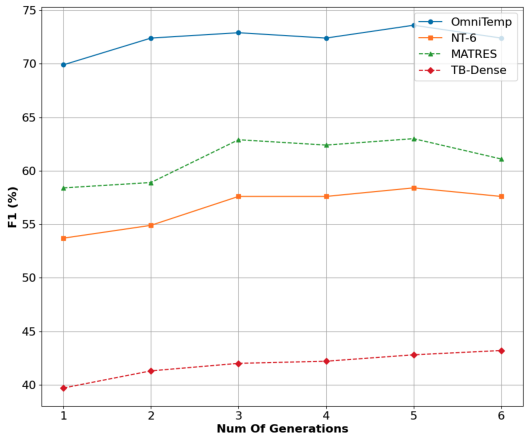



Figure 5: Effect of increasing the number of generated instances and applying GlobalConsistency with GPT-4o. Results show improved performance, with saturation observed after about five generations in most datasets.

| Model | NT-6 | OmniTemp |
|---|---|---|
| DeepSeek-R1 | 64.1 | 79.2 |
| DeepSeek-V3 | 55.1 | 74.4 |
| GPT-o3-mini | 55.5 | 78.1 |
| GPT-4o | 58.4 | 73.6 |
| Llama-3.1 405B Instruct | 42.6 | 57.9 |
| Llama 3.3 70B-Instruct | 40.7 | 44.3 |
| Gemini-flash 2.0 | 32.1 | 47.3 |

Table 5: Additional results of our GlobalConsistency approach when applying different LLMs, evaluated on the OmniTemp and NT-6 datasets.

| | GPT-4o | DeepSeek-R1 |
|---|---|---|
| GlobalConsistency | 71.2 | 64.9 |

Table 6: Results on the validation set of MAVEN-ERE (Wang et al., 2022) on the sub-set of relations we selected.

was run for 50 epochs on a single A100 GPU,[8] with the best-performing epoch on the development set selected for evaluation. For the GPT-4o experiments, we use 'gpt-4o-2024-08-06' version through OpenAI API, and used Together.ai API, for the DeepSeek-R1 experiments. We set the number of generations to five, based on tuning experiments with GPT-4o on the OmniTemp and NT-6 development sets (illustrated in Figure 5). In all experiments, we provide the model with all event pairs combinations, and evaluate on the available gold labels. For the MATRES and TimeBank-Dense (TB-Dense) datasets, we evenly divide the set of pairs in documents containing more than 20 events. In TB-Dense, for documents exceeding 40 events, we further group the pairs into sets of 100. Finally, In cases the generation missed pairs or is malformed, we regenerate the document or its respective split. For temporal constraint optimization, we employ the Gurobi Optimizer (Gurobi Optimization, LLC, 2024). Finally, the total experimental cost of this research—including CoT, ablation, and final results—using LLMs via OpenAI, Google, and Together.ai was approximately $400 (USD).

# B Additional Experiments

## B.1 Additional Tested LLMs

Beyond our main experiments with GPT-4o and DeepSeek-R1, we also evaluated our model with additional LLMs, summarized in Table 5 together with GPT-4o and DeepSeek-R1 for ease of comparison. GPT-o3-mini and DeepSeek-V3 achieved promising results on both NT-6 (55.1 and 55.5) and OmniTemp (74.4 and 78.1). Additionally, we tested our method with several contemporary models accessed via the Together.ai and Google Gemini APIs, including LLaMA (v3.1 405B, v3.3 70B) and Gemini (Flash-2.0, Pro-1.5). Our findings suggest that while all of these models (except Gemini Pro-1.5, which truncated the generation with the message: "rest of the pairs are similar, and the logic should follow the timeline explanation.") are capable of generating a complete set of temporal relations in a single step, they achieved much lower results. This indicates that our method for generating complete temporal graphs currently performs best with more advanced models.

## B.2 Additional Tested Dataset

We conducted an additional experiment with our GlobalConsistency method on the MAVEN-ERE (Wang et al., 2022) dataset, which introduces an additional domain, as it was curated from Wikipedia (in contrast to the newswire datasets used in our main experiments). MAVEN-ERE includes multiple relation categories, such as temporal, coreference, causal, and sub-events. In our setting, we used only the temporal relations portion of the dataset. For temporal relations, MAVEN-ERE defines six relation types: BEFORE, CONTAINS, OVERLAP, BEGINS-ON, ENDS-ON, and SIMULTANEOUS. To manage costs, we ran an experiment on the validation set, selecting docu-

[8] Experiment GPU time varies depending on the size of the training set, ranging from 1 to 20 hours for a full training episode.

ments with fewer than 200 pairs and considering only the BEFORE and SIMULTANEOUS relations. We applied the four-relations instruction (before, after, equal, and vague in our setting). The results are shown in Table 6, and we observe that the GlobalConsistency method achieves results comparable to those reported on the four-relation news datasets, further confirming the method's transferability across domains.

## C Further Main Result Table Details

In this section we provide further details on the results and measurements presented in Table 2

### C.1 Further Details on Reported Results

We provide further details on the results presented in Table 2. For the supervised models—RoBERTa, Bayesian, and Bayesian + Constraints—we report the best results achieved following a hyperparameter search (further detailed in Appendix A). For the CoT experiment, we conducted a single evaluation run for each dataset and used this result. Constructing an ensemble or computing the mean for this experiment across multiple runs was beyond our budget. In Table 7, we report the results for ZSL-Global and ZSL-Timeline, presenting the mean result obtained from five generations along with the standard deviation. For SelfConsistency and GlobalConsistency, we conducted a single run for each experiment, similar to CoT, as these experiments are more costly, and the observed standard deviation does not justify the additional expense.

### C.2 Transitive Inconsistency (TI) Details

For the Transitive Inconsistency (TI) measure reported in the table, we compute the average number of transitive-inconsistent edges per test document. We adopt a standard transitive closure algorithm (Warshall, 1962), which is typically used to construct transitive relations. In our case, for any inferred path that implies a transitive relation, we verify whether the resulting relation is among the set of transitively allowed relations, as defined in (Allen, 1984; Ning et al., 2018a) and related work. If the inferred relation violates this constraint, it is counted as a transitive inconsistency.

### C.3 Further Details on Event Count

In Figure 3, TB-Dense performance drops sharply for documents with over 25 events. Further analysis reveals that these documents predominantly contain *vague* relations—considered more challenging and often associated with annotator disagreement (Chambers et al., 2014). ZSL-Timeline struggles with these relations (Figure 4), particularly in TB-Dense and MATRES. As the frequency of *vague* relations decreases beyond this threshold, performance improves.

| Model | MATRES | TB-Dense | NT-6 | OmniTemp |
|---|---|---|---|---|
| ZSL-Global (Ours) | 59.0±1.4 | 37.7±1.8 | 48.4±2.5 | 62.3±0.5 |
| ZSL-Timeline (Ours) | 58.4±2.4 | 39.1±0.7 | 52.2±2.8 | 68.5±1.0 |

Table 7: F1 scores of ZSL-Global and ZSL-Timeline are reported along with the standard deviation.

| | Train | Dev | Test |
|---|---|---|---|
| MATRES | 13,577 | NA | 837 |
| TB-Dense | 4,205 | 649 | 1,451 |
| NarrativeTime | 68,317 | 2,759 | 7,925 |

Table 8: Statistics of event-event relations in the datasets used in this study.

## D Label Inconsistency Evaluation

We describe the *Label Inconsistency* experiment detailed in §7. MATRES (Ning et al., 2018b) and TB-Dense (Chambers et al., 2014) annotate the same set of 35 documents but follow different annotation schemes. MATRES considers only event start times to determine temporal order, while TB-Dense accounts for event start times, end times, and durations.

To isolate this difference, we define the following ground truth for each relation: (1) If a pair is marked as *vague* in MATRES, meaning the event start time is unclear, the same pair should also be *vague* in TB-Dense since both the start time and duration are uncertain. (2) If a pair in TB-Dense is annotated as *before*, *after*, or *equal* based on both start and end times, the corresponding MATRES annotation should reflect the same relation when considering only event start times. Figure 8 presents our findings in terms of label consistency and inconsistency between the two datasets.

## E OmniTemp Annotation Proccess

For the annotation process of OmniTemp (detailed in §3.1), we hired three annotators (two males and one female), all non-expert native English speakers and either undergraduate or graduate students. We instruct annotators to follow the MATRES annotation guidelines, considering only "actual" events (e.g., *they won the game*). Events that are "non-

actual", such as intentional, negated, recurring, conditional, or wishful (e.g., *I wish they win the game*), are excluded from annotation. Additionally, only the starting time of events is considered when establishing temporal relations.

The actual annotation was done on 30 news summaries, each containing approximately 500 words. The annotators used the EventFull annotation tool (Eirew et al., 2025), with all events in each document already highlighted. These events were extracted using the event detection method proposed by Cattan et al. (2021), which identifies all types of events (actual and non-actual) and extracts an average of 60 event mentions per document, forming the initial set of events. We follow the same annotation protocol as proposed in EventFull. First, the annotation process begins with the selection of 15 to 18 of the most salient "actual" events from each story, following Eirew et al. (2025) which found that beyond 18 events, annotation becomes challenging for non-expert annotators. This event reduction aligns with previous efforts to decrease annotation workload by limiting the number of events considered (Chambers et al., 2014; Ning et al., 2018b; Tan et al., 2024). After selecting these events, each document was annotated for temporal relations (*before*, *after*, *equal*, or *vague*) by all three annotators. Finally, majority voting was used to determine the final relation, and in cases of disagreement, the relation was labeled as *vague*.

Finally, the total annotation time for OmniTemp, including onboarding, amounted to 85 hours, with each worker paid $15 per hour (which is considered a fair market value in their region).

## F Formal Description of GlobalConsistency

GlobalConsistency is formulated as follows: we run the ZSL-Timeline method five times on each input as described in §4, generating five temporal graphs per document, denoted as $G = \{g_1, \ldots, g_5\}$ where each $g_n$ represents a labeled directed graph parsed from the DOT-language output. Each graph consists of a set of predicted event-pair relations: $g_n = \{p_{12}, p_{13}, \ldots, p_{23}, p_{24}, \ldots, p_{nm}\}$ where each relation $p_{ij}$ is represented as a one-hot vector over the six relation types. We then sum these vectors element-wise across all five graphs and normalize them to obtain a single distribution per event pair: $d_{ij} = \frac{1}{5}\sum_{n=1}^{5} p_{ij}^{(n)}$ where each $d_{ij}$ represents the normalized label distribution for the event pair $(e_i, e_j)$. Instead of selecting the most frequent relation via majority voting, we apply a temporal constraints optimization algorithm, which returns a temporally consistent graph. We call this final method GlobalConsistency (Figure 1).

To perform this optimization, we define a binary decision variable $\mathcal{I}_r(i,j) \in \{0,1\}$ for each relation $r \in \mathcal{R}$ and event pair $(e_i, e_j)$, where $\mathcal{R}$ is the set of possible temporal relations. The ILP objective is to maximize agreement with the model's predicted distributions:

$$\max_{\mathcal{I}} \sum_{i \neq j} \sum_{r \in \mathcal{R}} \mathcal{I}_r(i,j) \cdot d_{ij}^r$$

subject to the following constraints:

- **Uniqueness:** Each event pair must be assigned exactly one relation:

$$\sum_{r \in \mathcal{R}} \mathcal{I}_r(i,j) = 1 \quad \forall i \neq j$$

- **Symmetry:** For all inverse relations $r$ and $r^{-1}$, we ensure consistent labeling for reverse pairs:

$$\mathcal{I}_r(i,j) = \mathcal{I}_{r^{-1}}(j,i) \quad \forall r \in \mathcal{R}$$

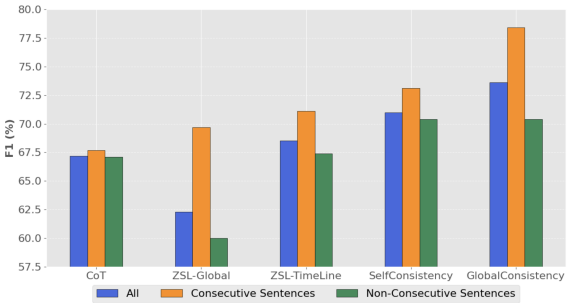
- **Transitivity:** For all event triplets $(e_i, e_j, e_k)$, if $\mathcal{I}_r(i,j) = 1$ and $\mathcal{I}_s(j,k) = 1$, then $\mathcal{I}_t(i,k) = 1$ for some $t \in \mathcal{C}(r,s)$, where $\mathcal{C}(r,s) \subseteq \mathcal{R}$ defines the transitive closure over $r$ and $s$, as specified in (Ning et al., 2018a).

This optimization ensures that the final output graph is both globally coherent and aligned with the model's confidence across multiple generations.

## G Dataset Licenses and Sources

In our experiments, we use the following commonly used datasets for evaluating the temporal relation extraction task: MATRES (Ning et al., 2018b), provided without a license; TimeBank-Dense (Chambers et al., 2014), provided without a license; and NarrativeTime (Rogers et al., 2024), provided under the MIT license. Additionally, OmniTemp uses summaries from the Multi-News corpus (Fabbri et al., 2019), which is distributed under a custom license that permits free academic use. All datasets were downloaded from official repositories, and used appropriately. OmniTemp will also be released under a free-to-use academic license.

(a) OmniTemp.




(b) NarrativeTime.

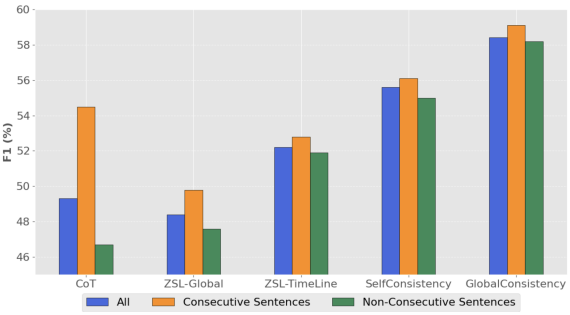
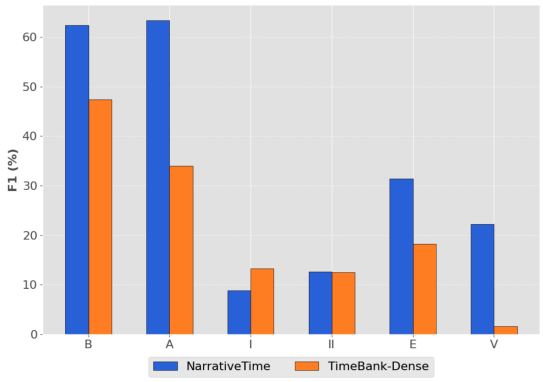



Figure 6: Similar to Figure 3, we examine the performance across different relation subsets for GPT-4o.

## H Adjustments to the NarrativeTime Dataset

The NarrativeTime (NT) dataset, introduced in §2.1, features seven relation types, including the six from TB-Dense and the *overlap* relation. Our temporal consistency algorithm relies on Allen's transitivity laws (Allen, 1984), which require each relation type to have a symmetric counterpart (e.g., if event *A* occurs *before* event *B*, then *B* must occur *after A*). However, the *overlap* relation in NT lacks a symmetric counterpart, making it incompatible for temporal consistency methods. Therefore, before using NT, we exclude event pairs labeled with the *overlap* relation. Additionally, NT documents contain an average of 50 event mentions per document, corresponding to approximately 1,100 relations, which makes them difficult to process with LLMs due to context length limitations. Handling such documents requires segmenting them and making individual calls to the model for each segment, which increases costs, as discussed in §4. To avoid segmentation and reduce costs, we randomly select 18 events per document from the test set, along with all their associated relations. The choice of 18 events was based on empirical observations, as it represents the maximum number that can typically fit within the model's context window without requiring segmentation. This reduction is not applied to the training set, which we use to fine-tune the supervised models. We refer to this pre-processed version as NT-6, as it retains only six relation types.

(a) NarrativeTime Vs. TimeBank-Dense.




(b) OmniTemp Vs. MATRES.

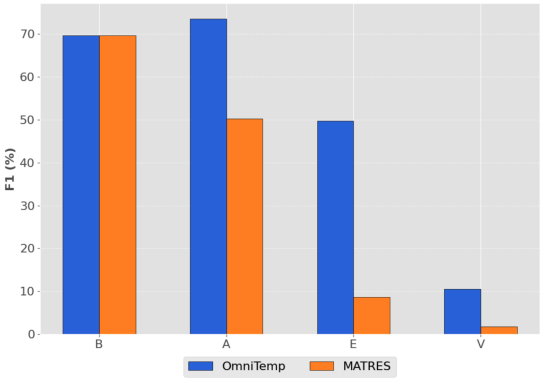
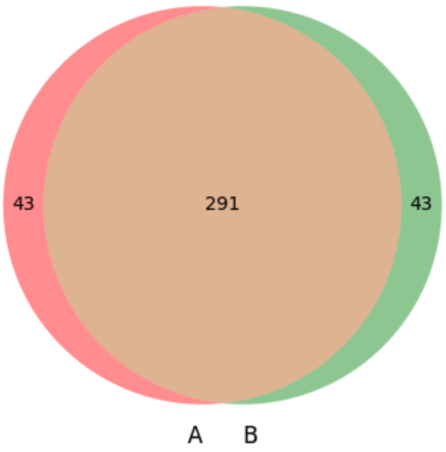


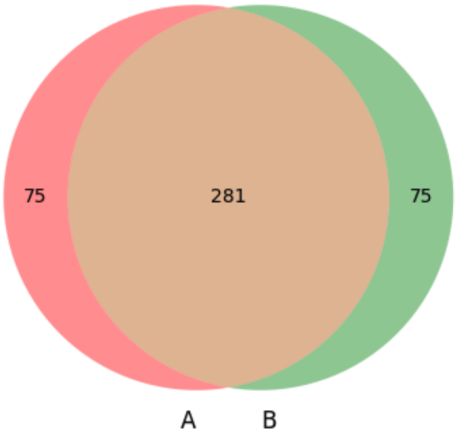

Figure 7: Similar to Figure 4, we examine the performance of our prompting method (i.e., ZSL-Timeline) by relation type using GPT-4o.

## I Additional Experiment Tables and Figures

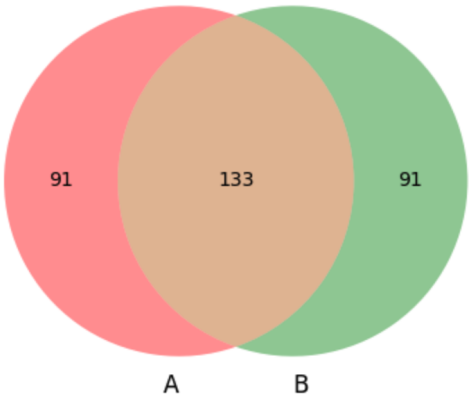
Figure 7 presents the relation-wise performance of GPT-4o, analogous to the results shown for DeepSeek-R1 in Figure 4. Figure 6 presents the model performance across different relation subsets, analogous to the results shown for DeepSeek-R1 in Figure 3. Table 9 presents a comparison between common datasets used for evaluating models on the temporal relation task alongside OmniTemp. Table 8 presents the split statistics of these datasets. Figure 9 presents an example of the ZSL-Global prompt. Figure 10 presents an example of the generated timeline using the ZSL-Timeline approach. Figure 11 presents the experimental results for filling transitive relations in a dataset containing only temporal relations between events up to one sentence apart (similar to MATRES and TB-Dense).



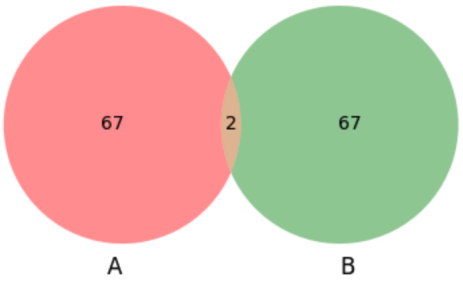

(a) Before




(b) After




(c) Vague




(d) Equal

Figure 8: Label Inconsistency: Each group, A and B, represents MATRES and TimeBank-Dense respectively. The intersecting area indicates consistency in label annotation between the two datasets, with the number of such pairs highlighted in the middle, while the non-intersecting areas represent pairs assigned different labels in each dataset.

Given the document below where each event is marked with <eventName(identifier)>, for each pair of events listed below, determine the temporal relationships (before, after, equal, vague) between them.
Answer in the following DOT format:

strict graph {
"Event1(id)" -- "Event2(id)" [rel=LABEL];
"Event1(id)" -- "Event3(id)" [rel=LABEL];
...
}
– Otto Warmbier is dead less than a week after <returning(39)> to the US from North Korea <The rest of the text is omitted to fit in Figure>...
Pairs require classification:
returning(39) -- detention(14)
returning(39) -- lived(3)
<The rest of the relations are omitted to fit in Figure>...

Figure 9: An example of the ZSL-Global prompt.

| | MATRES | TB-Dense | TCR | TDD-Manual | NarrativeTime | **OmniTemp** |
|---|---|---|---|---|---|---|
| **Datasets Statistics** | | | | | | |
| Documents | 275 | 36 | 25 | 34 | 36 | 30 |
| Events | 6,099 | 1,498 | 1,134 | 1,101 | 1,715 | 470 |
| *before* | 6,852 (50) | 1,361 (21) | 1,780 (67) | 1,561 (25) | 17,011 (22) | 1,540 (44) |
| *after* | 4,752 (35) | 1,182 (19) | 862 (33) | 1,054 (17) | 18,366 (23) | 1,347 (39) |
| *equal* | 448 (4) | 237 (4) | 4 (0) | 140 (2) | 5,298 (7) | 150 (4) |
| *vague* | 1,525 (11) | 2,837 (45) | – | – | 25,679 (33) | 446 (13) |
| *includes* | – | 305 (5) | – | 2,008 (33) | 5,781 (7) | – |
| *is-included* | – | 383 (6) | – | 1,387 (23) | 6,639 (8) | – |
| *overlaps* | – | – | – | – | 227 (0) | – |
| Total Relations | 13,577 | 6,305 | 2,646 | 6,150 | 79,001 | 3,483 |
| **Per Document Average Annotation Sparsity** | | | | | | |
| Events | 22.2 | 41.6 | 45.4 | 32.4 | 47.6 | 15.6 |
| Actual Relations | 49.4 | 183.7 | 105.8 | 180.9 | 1,110.1 | 114.9 |
| Expected Relations | 234.8 | 844.5 | 1,006.1 | 508.1 | 1,110.1 | 114.9 |
| Missing Relations | 79% | 78.3% | 89.5% | 64.4% | 0% | 0% |

Table 9: The upper part of the table presents the statistics of notable datasets for the temporal relation extraction task alongside OmniTemp. In parentheses, the values indicate the percentage of each relation type relative to the total relations in the dataset. The bottom part of the table summarizes the average percentage of missing relations per document, calculated as the ratio of actual annotated relations to a complete relation coverage, referred to as *Expected Relations*.

### Explanation of the Story Timeline

The timeline of events regarding the downing of Malaysia Airlines Flight 17 and subsequent developments is as follows:

1. **Downing of the Flight (downing(6), shot(24), fired(8))**: The initial event is the shooting down of Malaysia Airlines Flight 17 over eastern Ukraine. This was carried out by a Soviet-era SA-11 Buk missile, which was fired (fired(8)), resulting in the plane being shot down (shot(24)).
2. **Rebel Reaction (realize(3), realize(3), react(44))**: Following the downing, pro-Russia rebels reportedly realized that they had not shot a military target but rather a civilian aircraft. This realization led to a reaction of horror (react(44)) among them.
3. **Rebel Bragging (brag(33))**: An intercepted call was released (released(25)) by Ukraine's security services where pro-Russia rebels initially bragged (brag(33)) about shooting down a plane.
4. **Release of Recordings (released(25), recordings(21))**: Ukraine's security services released (released(25)) these recordings to demonstrate the rebels' initial stance before they realized the gravity of their action.
5. **Responses from World Leaders (response(14))**: Following the incident and subsequent released recordings, global leaders, including the Australian Prime Minister, conveyed strong dissatisfaction with Russia's response to this tragic event.
6. **Search and Recovery at Crash Site (search(16), wreckage(10), recovered(28))**: In the aftermath, recovery operations commenced at the crash site. Search teams looked for bodies (search(16)) among the wreckage (wreckage(10)), and by this time over 100 bodies had been recovered (recovered(28)).
7. **Rerouting and Avoiding the Airspace (rerouting(37), avoid(26))**: In response to the tragedy, airlines began rerouting (rerouting(37)) their flights to avoid (avoid(26)) the region's airspace due to security concerns.
8. **Rebels Finding the Black Boxes (found(45))**: The rebels found (found(45)) the aircraft's black boxes, which has significant implications for the investigation.
9. **Development of Crash Investigation (development(43), investigation(11))**: Concerns arose about the integrity of the investigation due to the rebels' control over the area and their possession of the black boxes.
10. **Blame Game (blaming(7))**: Russia and Ukraine started blaming (blaming(7)) each other for the tragedy. Both agreed on the weapon used but differed on who was responsible.

### Temporal Relationships in DOT Format

Based on the timeline explanation, here are the temporal relationships between the events in DOT format:

```
dot strict graph {
    \"response(14)\" -- \"downing(6)\" [rel=after];
    \"response(14)\" -- \"released(25)\" [rel=after];
    ...
}```
```

Figure 10: An example of a generated output when GPT-4o is prompted using the ZSL-Timeline method (with the Markdown format retained from the original output). The full event list is generated; however, it is trimmed (indicated by "...") in this example to ensure the output fits within the figure.

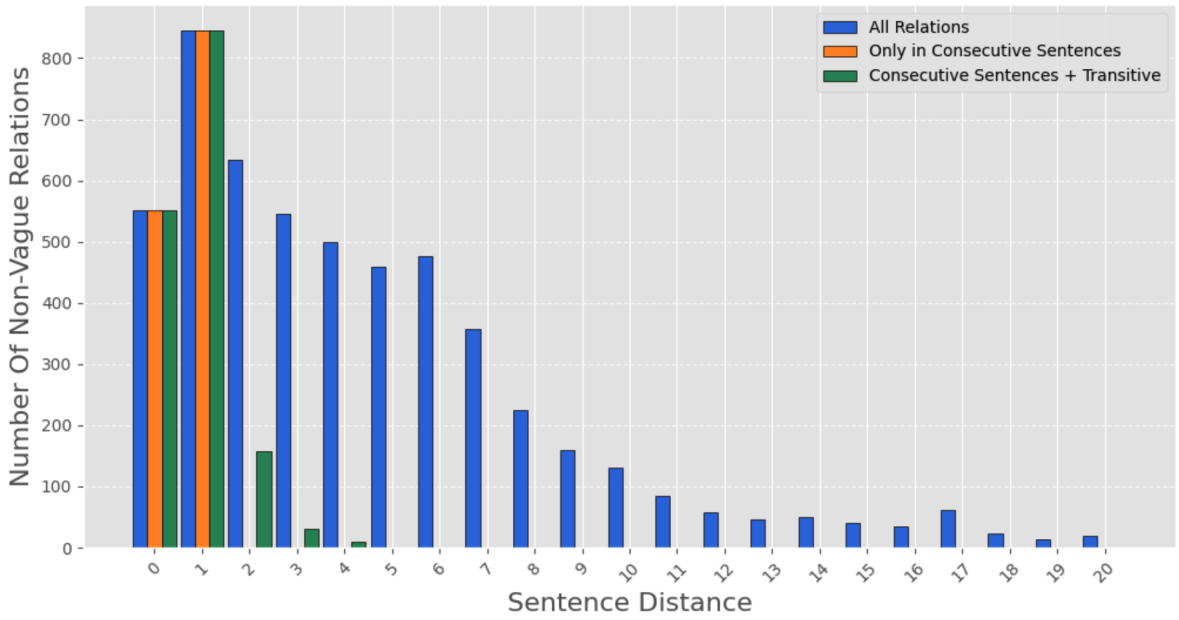



Figure 11: Illustration of the achieved relation distance after applying transitive closure in resources annotated only between consecutive sentences. The blue bars represent the original set of relations in NarrativeTime, which is exhaustively annotated between all events. The orange bars represent the version created by considering only relations between events in consecutive sentences. The green bars represent the set of relations after applying a transitive algorithm to infer additional relations.