LEAF: Linguistically Enhanced Event Temporal Relation Framework

Stanley Lim Da Yin Nanyun Peng
University of California, Los Angeles
stanleylim2@g.ucla.edu {da.yin, violetpeng}@cs.ucla.edu

Abstract

Linguistic structures can implicitly imply diverse types of event relations that have been previously underexplored. For example, the sentence "John was **cooking** freshly **made** noodles for the family **gathering**" contains no explicit temporal indicators between the events, such as *before*. Despite this, it is easy for humans to conclude, based on syntax, that the noodles were **made** *before* John started **cooking**, and that the family **gathering** starts *after* John starts **cooking**. We introduce **L**inguistically enhanced **E**vent Tempor**A**l relation **F**ramework (**LEAF**), a simple and effective approach to acquiring rich temporal knowledge of events from large-scale corpora. This method improves pre-trained language models by automatically extracting temporal relation knowledge from unannotated corpora using diverse temporal knowledge patterns. We begin by manually curating a comprehensive list of atomic patterns that imply temporal relations between events. These patterns involve event pairs in which one event is contained within the argument of the other. Using transitivity, we discover compositional patterns and assign labels to event pairs involving these patterns. Finally, we make language models learn the rich knowledge by pre-training with the acquired temporal relation supervision. Experiments show that our method outperforms or rivals previous models on two event relation datasets: MATRES and TB-Dense. Our approach is also simpler from past works and excels at identifying complex compositional event relations.

1 Introduction

Event temporal relation extraction can help us better organize event flow and understand how events develop. For example, in news articles, understanding the causal relationships between events can help us better understand why certain events occurred (Tan et al., 2022; Zhang et al., 2023). In medical records, understanding the temporal relationships between events can help us better track a patient's medical history (Cheng et al., 2013; Lee et al., 2018).

Recently, there have been works focusing on first acquiring temporal relation knowledge automatically and then injecting the acquired knowledge via pre-training. For example, ECONET (Han et al., 2021b) uses explicit keyword search to retrieve the sentences that contain temporal indicators such as *before*, *after*, *during*, and *previously* as supervision. However, they do not fully exploit knowledge from sentence **linguistic structures**. While Zhou et al. 2020a make an attempt to utilize linguistic structures by extracting patterns from semantic role labeling (SRL) parses (Gardner et al., 2018; Shi and Lin, 2019), much of the linguistic information available is under-explored and they only utilize keywords found in an event's temporal argument. Moreover, previous works which utilize linguistic structure for event relation knowledge do not apply them to neural networks (D'Souza and Ng, 2013; Chambers et al., 2014).

We find that there is rich, implicit event knowledge in the **linguistic structures** that has not been explicitly leveraged in the past. For example, consider the sentence "John was **cooking** freshly **made** noodles for the family **gathering** as Adam **arrived**". Notice that in this sentence, there is only one ***explicit*** mention of the temporal relationship between any of the events, which is that John was **cooking** as Adam **arrived**. However, it is possible to extract five times the number of relations according to linguistic structures (Figure 1):

Relation 1: The noodles were **made** *before* John started **cooking**, since adjectives of event objects all occur before the event occurs.

Relation 2: John started **cooking** *before* family **gathering** began, since an event always starts before its purpose event.

Relation 3: The noodles were **made** *before* Adam **arrived**, since we know from the above relations that the noodles were made before John

Proceedings of the 2nd Workshop in Pattern-based Approaches to NLP in the Age of Deep Learning , pages 6–19
December 6, 2023. ©2023 Association for Computational Linguistics

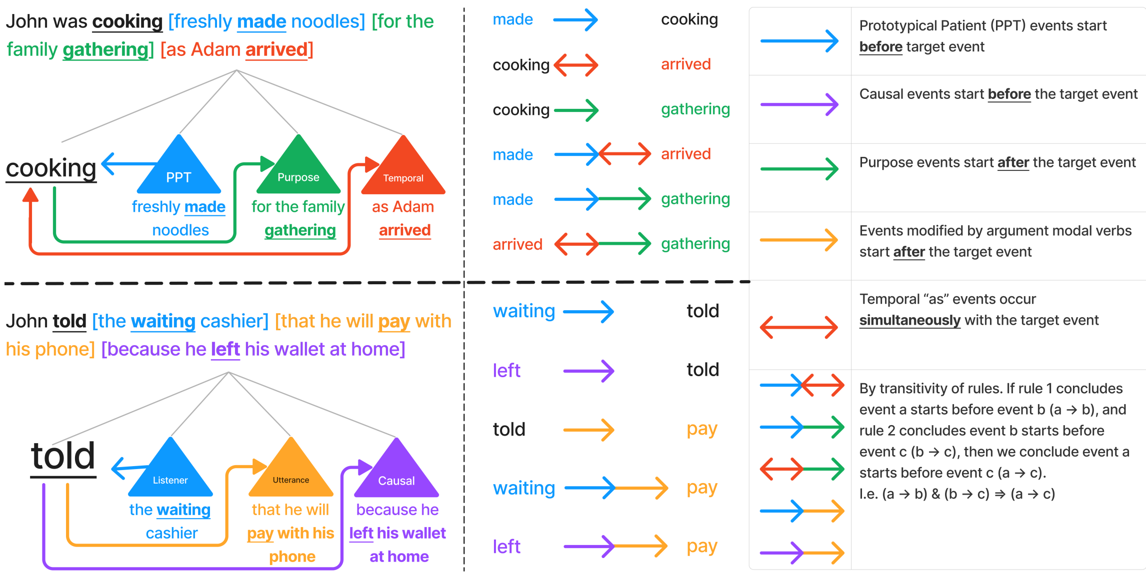

Figure 1: Process of utilizing temporal knowledge patterns to acquire temporal relation supervision for pre-training. An arrow a –> b indicates that event "a" starts before event "b". We first obtain the SRL annotations for all events (shown are the annotations for "cooking" and "told"). Then, using a list of atomic patterns, we automatically extract temporal relationships between a target event and other events in its arguments (shown by the single colored arrows). Finally, we use transitivity rules to find compositional relations (shown by the arrows with two colors). The table on the right shows the patterns corresponding to each relation. Note that this figure displays a subset of the entire set of patterns we use.

started **cooking**, and Adam **arrived** as cooking was started.

Relation 4: The noodles were **made** *before* the family **gathering**, since we know from the above relations that the noodles were **made** before John began **cooking**, and John began **cooking** before family **gathering** started.

Relation 5: Adam **arrived** *before* family **gathering**, since we know from the above relations that Adam **arrived** as John was **cooking**, and John starts **cooking** before the family **gathering**.

To this end, we propose **LEAF**, a **L**inguistically enhanced **E**vent tempor**A**l relation **F**ramework. Our method aims at capturing a diverse set of linguistic structures implicitly indicating temporal relations (a.k.a, temporal knowledge patterns), and uses them to facilitate language models to learn richer temporal knowledge. We start by manually curating a diverse list of **atomic** patterns that commonly imply certain temporal relations (Appendix A). These patterns involve pairs of events where one event is contained within the argument structure of the other. For example, a target event always starts after events in its prototypical patient (PPT) argument, and we can use this atomic pattern to find that **cooking** starts after **made** (Figure 1). Our list encompasses an extension of patterns from previous works (Zhou et al., 2020a) along with novel patterns, including the PPT pattern.

As illustrated in Figure 1, if we consider only atomic patterns, many temporal relations are still overlooked. The events **made**, **gathering**, and **arrived** are not within one another's arguments, yet they still hold temporal relations. To capture these relations, we also gather **compositional** patterns. These are connections between two events that are not directly linked in their argument structures and are derived by utilizing the transitivity of atomic patterns. From applying these temporal knowledge patterns, we are able to extract two more atomic relations (relations 1-2) and three more compositional relations (relations 3-5) from the example sentence than knowledge acquisition methods relying only on temporal indicator word searching.

To deploy temporal knowledge patterns for obtaining training supervision at scale, we use AllenNLP's SRL parser (Gardner et al., 2018; Shi and Lin, 2019) on raw text. We then search the collected patterns to determine if any patterns appear in the SRL annotations of a given sentence. Once a pattern is found, the corresponding temporal relation of the pattern can then be used as supervision to further pre-train language models. With this method, we collect millions of event relation supervisions for pre-training from the raw Gigaword headline corpus (Graff et al., 2003; Rush et al., 2015). This includes around 3.8M atomic relations and 140K compositional relations.

Our method effectively helps pre-trained language models learn rich temporal knowledge. LEAF demonstrates an improvement of up to 9 F1 over vanilla BERT$_{BASE}$ and RoBERTa$_{BASE}$ on MATRES (Ning et al., 2018) and TB-Dense (Cassidy et al., 2014). It also delivers competitive performance with previous state-of-the-art (SOTA) methods that use temporal indicators and complex fine-tuning layers. Moreover, it greatly exceeds 3-shot ChatGPT (OpenAI, 2023) by over 37 F1 points. We also perform ablation studies, which verify that both types of temporal knowledge patterns contribute to high performance. Finally, with the aid of acquired atomic and compositional relation supervision, LEAF shows an increase of up to 6.8 F1 points over baselines on challenging cases involving compositional relation prediction.

2 Related Work

In the early stages of event relation research, experts often used traditional machine learning methods to classify relations (Chklovski and Pantel, 2004; Mani et al., 2006; Pitler and Nenkova, 2009; Mirza, 2014). These methods required experts to manually identify features and use external resources, which was time-consuming and labor-intensive. Recently, there have emerged a great number of attempts to incorporate temporal relation knowledge into neural network models (Cheng and Miyao, 2017; Goyal and Durrett, 2019; Xie et al., 2022). One branch of this involves incorporating additional temporal knowledge in the fine-tuning stage on fully labeled datasets, then evaluating on the respective dataset. Some add additional parameters to train (Tan et al., 2021; Hwang et al., 2022; Lu et al., 2022; Wen and Ji, 2021), while others only add objectives during fine-tuning (Wang et al., 2022a; Zhang et al., 2022).

Another branch called weak supervision is more closely related to our work. Weak supervision does not require expensive manually labeled training data, but instead automatically labels unannotated corpora (Xie et al., 2022). This allows for greater transferability of knowledge between tasks, as well as ease of scalability. There are several popular methods for extracting event temporal relations using weakly supervised data. One approach is to perform a keyword search (Zhao et al., 2021; Han et al., 2021b). Another approach is to use a teacher model to label event relations (Ballesteros et al., 2020). The most closely related to our work is Zhou et al. 2020a, which uses keywords within a single SRL semantic tag to extract relational knowledge. However, this is only a small subset of all available syntactic knowledge. In this work, we expand the range of syntactic structures used to extract relation knowledge, enabling language models to learn more diverse and complex knowledge.

3 Method

3.1 Overview

In this section, we describe our method for extracting temporal relation knowledge from unlabeled data and continually pre-train a language model to inject this knowledge. We begin by providing background on syntactic and semantic terminology (§3.2). Next, we offer a precise definition of atomic patterns and expand on the original patterns discovered through our work (§3.3.1). We then discuss the process of culminating compositional patterns and their importance (§3.3.2). In §3.4, we detail how we obtain temporal relation supervision with our collected patterns. Finally, we show how to use the relation supervision to pre-train language models (§3.5).

3.2 Background of Syntactic and Semantic Terminology

In this section, we introduce background on syntactic and semantic terminology involved in LEAF.

An event refers to a specific occurrence of something that happens in a certain time and a certain place involving one or more participants, which can usually be described as a change of state (Li et al., 2022). Following previous works, we define the relation between two events by the occurrence of their start time (Ning et al., 2018). Consider the sentence "John was **cooking** freshly **made** noodles for the family **gathering**" and its two events e_1 = **cooking** and e_2 = **made**. In this sentence, **cooking** clearly starts after **made**, so we would label the relation between e_1 and e_2 as "after". Specifically, we consider three temporal relationships for the temporal relation supervision in pre-training stage: *before, after,* and *simultaneous*.

Verbs are elements that encode events and hold arguments. For example, in the sentence "John was **cooking** freshly **made** noodles for the family

gathering as Adam **arrived**," the verb **cooking** takes four semantic arguments: agent, prototypical patient (PPT), purpose (PRP), and temporal (TMP) (Figure 2). "John" is the agent, "freshly made noodles" is the PPT, "for the family gathering" is the PRP argument, and "as Adam arrived" is the TMP argument. Arguments are the key components of our collected temporal knowledge patterns introduced in §3.3.

3.3 Temporal Knowledge Patterns

Temporal knowledge patterns are linguistic structures which usually imply certain temporal relations. The goal in collecting patterns is to extract rich event temporal knowledge from unlabeled text, allowing for harvesting of large-scale pre-training supervision. In this section, we introduce how we curate a diverse suite of atomic and compositional patterns, covering a vast range of linguistic information.

3.3.1 Atomic Patterns

Atomic patterns involve pairs of events where one event is contained within the argument structure of the other. Take the example "John was **cooking** freshly **made** noodles for the family **gathering** as Adam **arrived**." Since **made**, **gathering**, and **arrived** are all within **cooking**'s argument, atomic patterns may underlie the linguistic structures between **cooking** and the other three events (Figure 1). To curate atomic patterns that likely indicate certain temporal relations, we analyze examples from existing temporal relation datasets (Han et al., 2021a; Ning et al., 2020; Wang et al., 2022b). A subset of the atomic pattern list can be found in Table 1, and the comprehensive list can be found in Appendix A. Our list of atomic patterns includes both extensions of patterns explored in previous works (Zhou et al., 2020a) and novel patterns with linguistic structures implicitly expressing temporal relations.

One example of an atomic pattern that does not make use of any explicit temporal indicators is the prototypical patient (PPT) modifier pattern. A PPT is an event argument that undergoes change or is affected by the target event. We find that events which modify the PPT of a target event start before the respective target event. In the sentence previously mentioned, after observing that the PPT of **cooking** is "freshly **made** noodles," we can use this pattern to extract the relation that **made** starts before **cooking** (Figure 2). As events are commonly accompanied by PPT arguments in everyday English, detecting PPT patterns help obtain abundant temporal relation supervision for further pre-training.

The general PRP tag pattern and the general CAU tag pattern are two other examples of atomic patterns which do not use explicit temporal indicators. Both of these require no keyword occurrences and we can easily detect them with SRL tools. Two examples are shown in Figure 1.

3.3.2 Compositional Patterns

As displayed in Figure 1, many event pairs do not appear in each other's arguments, and thus their relationships cannot be concluded with just atomic patterns. For example, in the sentence "John was **cooking** freshly **made** noodles for the family **gathering** as Adam **arrived**," none of **made**, **gathering**, or **arrived** are in each other's arguments, yet there still exists temporal relations between them. Compositional patterns involve pairs of such events that are not in each other's arguments. These patterns are higher in difficulty than atomic patterns, as the events are more loosely connected with each other according to the syntax. Previous works have explored only atomic relations between two events to provide supervision (Zhou et al., 2020a; Han et al., 2021b), without considering compositional patterns. We value the importance of compositional temporal relational knowledge and further leverage it as sources of additional supervision to better tackle the challenges. This allows us to not only capture more linguistically complex relations, but also inter-sentence relations.

Compositional patterns frequently appear under the following circumstance: consider three events e_1, e_2, e_3, where e_1 is not in e_3's arguments, e_3 is not in e_1's arguments, but e_2 is in both e_1 and e_3's arguments. In a scenario where atomic patterns find that e_1 occurs before e_2, and e_2 occurs before e_3, then we can also utilize transitivity to conclude e_1 occurs before e_3 without e_3 being in any of e_1's arguments and e_1 being in any of e_3's arguments.

We cumulate a comprehensive list of all possible compositional patterns that can result from transitivity between two atomic patterns. Then, we use these compositional patterns to extract compositional relations from SRL annotated text.

3.4 Creating Supervision From Patterns

In this section, we detail the algorithm we deploy in order to extract temporal relations from raw text.

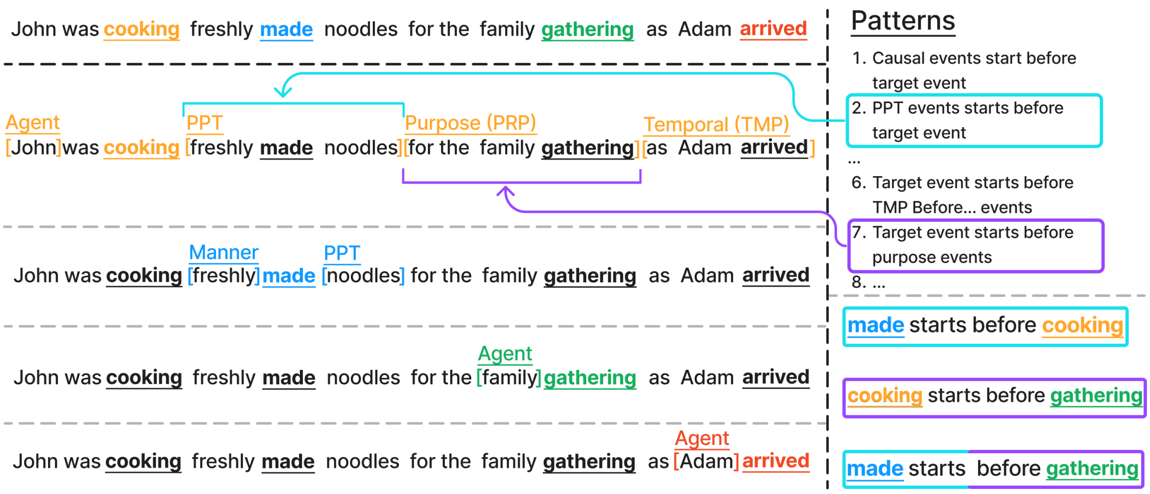

Figure 2: Examples to acquire temporal relation supervision via pattern matching. From a single sentence (top-left), we get the semantic arguments for each event using the SRL parser. The PPT pattern can help extract the relation that "made" starts before "cooking" (cyan), and the PRP pattern is used to conclude that "cooking" starts before "gathering" (purple). Finally, we use compositional patterns to gather complex relationships. We use the transitivity of the two atomic patterns to extract that "made" starts before "gathering" (cyan + purple box).

Names	Temporal Relations	Example Sentences	Explanations
CAU	After	John cooked noodles [because he was hungry].	John cooked after he was hungry
PRP	Before	John cooked noodles [for the family gathering].	John cooked before the family gathering
PPT	After	John cooked [freshly made noodles].	The noodles were made before John cooked them

Table 1: Subset of atomic patterns. CAU and PRP correspond to events in the causal and purpose tag, respectively. The PPT row refers to prototypical patient tags. These patterns indicate that all events in that tag hold the respective temporal relation to the target event.

We begin with the unannotated Gigaword headlines corpus[1] (Graff et al., 2003; Rush et al., 2015), which consists of around 3.8M news headline sentences. Then, we obtain SRL annotations of the headline sentences with SRL parser. The parser provides all arguments for each event within a headline. For example, in the sentence in Figure 2, each event **made**, **cooking**, **gathering**, and **arrived** will have its arguments labeled. Concretely, we use AllenNLP semantic role labeling (SRL) parser (Gardner et al., 2018; Shi and Lin, 2019) to obtain detailed annotation of events and the specific roles of their arguments.

With each event's arguments annotated by SRL tools, we iterate through each event and detect the existence of temporal knowledge patterns (§3.3) in each sentence. Specifically, given a target event, we first examine whether there are any atomic patterns underlying the linguistic structure of the given texts. If there is, we are able to extract relations between the target event and the events within its arguments. For example, PPT pattern is detected and helps us extract the relation that **made** starts before cooking. The PRP pattern can also be found to conclude that **cooking** starts before **gathering**. Finally, we use our list of compositional patterns to extract compositional relations between events which do not occur within each other's arguments. The process of obtaining compositional relations must also follow the principle of temporal relation transitivity. A compositional relation that we extract in the above sentence is that **made** starts before **gathering**, by transitivity of the two atomic relations above (Figure 2).

In total, we extract 3.8M atomic relations and 140K compositional relations with our collected temporal knowledge patterns. The extracted relations are all treated as the temporal relation supervision for later pre-training. To verify the accuracy of the acquired supervision, we got 2 undergraduate students to annotate 50 instances each. We observe that 86% of the sampled instances are correct. This indicates the reliability of our process

[1] https://huggingface.co/datasets/gigaword

for automatically acquiring supervision.

3.5 Pre-training LMs with Acquired Temporal Knowledge

In this section, we introduce our pre-training method to make LMs learn rich temporal knowledge with our acquired relation supervision.

Specifically, we adopt BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) as our base models and initialize the models with their pre-trained parameters. The masked language modeling (MLM) objective is one of our leveraged pre-training objectives. Suppose that there is an input sequence $X = [x_1, x_2, ..., x_n]$, where x_i indicates the token at the i-th index. The pre-training objective is to minimize the negative log-likelihood of predicting the masked tokens given the contexts. Thus, $\mathcal{L}_{MLM}$ is the cross-entropy loss value of predicting the masked tokens. We use the traditional BERT masking technique where 15% of tokens are either masked, replaced with a random token, or left unchanged (Devlin et al., 2019). Additionally, following previous work, we target 25% of event tokens for one of these transformations (Han et al., 2021b; Zhou et al., 2020a; Kimura et al., 2022).

Along with the traditional MLM pre-training objective, to let our models learn the acquired temporal knowledge, we utilize a temporal relation prediction objective (Ballesteros et al., 2020; Wang et al., 2020) as the other pre-training objective. Consider the contextualized embeddings $\mathcal{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n]$ obtained from our base models. Let $\mathbf{h}_i$, $\mathbf{h}_j$ be the contextualized representations for the tokens of e_1 and e_2, respectively[2], and let $\mathbf{g}_p$, $\mathbf{g}_s$ be their element-wise Hadamard product and subtraction (Zhou et al., 2020b; Wang et al., 2020). We then feed $[\mathbf{h}_i : \mathbf{h}_j : \mathbf{g}_p : \mathbf{g}_s]$ into a multi-class classifier, where each class corresponds to one of the three considered temporal relations *before, after,* and *simultaneous*, to obtain $\hat{\mathbf{y}}$. We define the temporal relation objective $\mathcal{L}_{REL}$ as:

$$\mathcal{L}_{REL} = -\frac{1}{m}\sum_{i=1}^{m} \mathbf{y}_i \log(softmax(\hat{\mathbf{y}}_i)), \quad (1)$$

where $\mathbf{y}$ is the one-hot ground-truth vector and m is the number of training instances.

Our final loss function is thus:

$$\mathcal{L} = \mathcal{L}_{MLM} + \mathcal{L}_{REL}. \quad (2)$$

[2]For events that span multiple tokens, we simply take the first token of the event as the representation.

4 Experiments

In this section, we present experiments to demonstrate the effectiveness of LEAF for acquiring rich temporal relation knowledge. LEAF is capable of assisting LMs achieve high performance on multiple downstream event relation benchmarks. It facilitates base models to perform comparable with previous SOTA models (Section 4.2). We also verify the significance of pre-training objectives and data by conducting ablation studies (Section 4.3). Next, we reveal our method's effectiveness at predicting compositional relations (Section 4.4). Finally, we perform a case study to analyze LEAF's effectiveness at learning patterns that were seen and those that were unseen during pre-training (Section 4.5).

4.1 Experimental Setup

For our experiments, we pre-train both BERT$_{BASE}$ and RoBERTa$_{BASE}$ on our extracted data (Devlin et al., 2019; Liu et al., 2019). We train on 4 GeForce GTX 1080 Ti's for 3 epochs. For BERT$_{BASE}$, we use a 1e-4 learning rate, 0.2 dropout rate, and a batch size of 32. For RoBERTa$_{BASE}$, we use a 5e-5 learning rate, 0.3 dropout rate, and a batch size of 24.

For evaluation, we consider two temporal relation extraction datasets: MATRES (Ning et al., 2018) and TB-Dense (Cassidy et al., 2014). Details for the datasets can be found in Appendix B. For both datasets, we train a new classifier head with m output dimensions, where m is the number of labels of the respective dataset. We fine-tune for 10 epochs on both datasets, and following previous works, we report the micro-F1 score for each dataset (Wang et al., 2020).

4.2 Comparisons with Existing Systems

We compare our proposed method with two base models BERT and RoBERTa. We also demonstrate the effectiveness of LEAF in comparison with previous SOTA models.

4.2.1 Base-Models: BERT and RoBERTa

To evaluate the effectiveness of our method, we compare the performance of LEAF-enhanced BERT$_{BASE}$ and RoBERTa$_{BASE}$ with their respective vanilla counterparts. The results in Table 2 show that LEAF-enhanced models outperform the vanilla models on both datasets by a large margin. In particular, LEAF can bring 9-11 F1 improvement over vanilla RoBERTa$_{BASE}$. This demon-

	MATRES	TB-Dense
BERT$_{BASE}$	73.7	58.1
+ LEAF	81.3	63.2
RoBERTa$_{BASE}$	73.1	55.7
+ LEAF	**82.1**	66.7
ChatGPT(0-shot)	26.2	22.0
ChatGPT(3-shot)	49.2	29.1
Bi-LSTM (Cheng and Miyao, 2017)	59.5	48.4
TacoLM^{+} (Zhou et al., 2020a)	63.5	40.1
Goyal and Durrett (2019)	68.6	—
BERE-p (Hwang et al., 2022)	71.1	—
EventPlus (Ma et al., 2021)	75.5	64.5
SP+ILP (Ning et al., 2017)	76.3	58.4
Wang et al. (2020)	78.8	—
Poincaré Event Embeddings (Tan et al., 2021)	78.9	—
United-Framework (base) (Huang et al., 2023)	79.3	66.4
ECONET (Han et al., 2021b)	79.3	**66.8**
HGRU+knowledge (Tan et al., 2021)	80.5	—
Ballesteros et al. (2020)	81.6	—
Wen and Ji (2021)	81.7	—

Table 2: Overall experimental results. Following previous works, we report micro-F1 score for both datasets. $^{+}$ denotes our reproduced results. Note that ECONET is based on RoBERTa$_{LARGE}$, which is 3× bigger than our base models. We still outperform ECONET on MATRES by a large margin.

strates the benefits of pre-training with rich temporal knowledge acquired with LEAF methods.

4.2.2 Previous SOTA Models

We also compare LEAF method to 12 previous SOTA models, and find that LEAF leads to competitive performance on both datasets. Along with being simpler in design, our method requires training no additional parameters beyond a classifier, and outperforms other models with over triple the parameters. This includes outperforming EventPlus (Ma et al., 2021), a pipeline which uses twice the parameters of our model, by 6.6 F1. Wen and Ji (2021) and ECONET (Han et al., 2021b) are based on RoBERTa$_{LARGE}$, which is 3 times larger than RoBERTa$_{BASE}$. Nevertheless, RoBERTa$_{BASE}$ + LEAF surpasses both models as well.

4.2.3 Models Relying Only on Explicit Temporal Indicators

LEAF focuses on capturing richer temporal knowledge **implicitly** expressed in texts. In contrast, previous works about temporal knowledge acquisition merely utilize **explicit** indicators when gathering temporal knowledge from text. In this section, we verify the importance of implicit indicators by comparing our model to those that do not utilize this extra information when curating temporal patterns. The two models that we compare with in this section are ECONET and TacoLM.

In order to automatically gather temporal information for supervision, ECONET (Han et al., 2021b) collects a list of keywords that each imply a certain temporal relationship. For example, the words "before, until, and preceding" all imply the same temporal relation between events. However, they ignore crucial linguistic information by only doing keyword search for their patterns, limiting their scope to explicitly stated temporal relations. Results in Table 2 show that although the base model of ECONET is RoBERTa$_{LARGE}$ which is 3× bigger than our base models, LEAF can still outperform ECONET by 2.8 F1 on MATRES and achieve nearly the same performance on TB-Dense.

The major limiting factor of TacoLM discussed in §3.3.1 is that they only use a small subset of linguistic information to extract their temporal relation knowledge. In particular, they only consider the temporal arguments of events when acquiring temporal relation supervision. We conduct experiments to reproduce TacoLM based on BERT$_{BASE}$ and then evaluate the model on these two datasets. Results are shown in Table 2. Despite being trained on ~21M data, TacoLM underperforms BERT$_{BASE}$ + LEAF by a large margin.

4.2.4 ChatGPT

In this section, we analyze the performance of ChatGPT (`gpt-3.5-turbo` on 05-20-2023) on the two downstream datasets. We first design three different prompts, and for each prompt, we have a zero-shot and a three-shot variant, totalling six prompts per evaluation task (Appendix C). We then evaluate ChatGPT on TB-Dense and MATRES. Results can be found in Table 2. Aligning with past findings (Kauf et al., 2022; Yuan et al., 2023), we observe that ChatGPT does poorly at identifying event temporal relations. Both the 3-shot and the zero-shot F1 scores are significantly worse than BERT$_{BASE}$ + LEAF.

4.3 Ablations

Ablation study for pre-training objectives. To verify the significance of our pre-training objectives towards the better model performance, we conduct ablation studies to examine the effect of removing MLM and temporal relationship prediction objectives. Results can be found in Table 3. We find that for both datasets, removing either MLM or temporal relationship prediction objective leads to a worse performance than pre-training with both objectives. This indicates that both objectives are crucial in allowing the model to learn temporal

	MATRES	TB-Dense
BERT$_{BASE}$	73.7	58.1
+ ***LEAF***	**81.3**	**63.2**
- ***TMP REL***	80.2	62.0
- ***MLM***	56.6	29.7
- Atomic	79.6	60.7
- Compositional	79.9	59.9

Table 3: Ablation studies of the training objectives and patterns. The addition of LEAF improves the performance of BERT$_{BASE}$ on each dataset. The combination of the two training objectives is effective, as removing either one lowers performance on the two datasets. The combination of both types of temporal knowledge patterns also proves to be crucial, as removing either one also lowers performance on both datasets.

	MATRES-C	TB-Dense-C
BERT$_{BASE}$	72.8	53.2
+ ***LEAF***	78.5	57.0
- Atomic	81.0†	60.1†
- Compositional	75.1	53.1
RoBERTa$_{BASE}$	74.6	55.2
+ ***LEAF***	**81.1**	**62.0**

Table 4: Results on the instances involving compositional relations in MATRES and TB-Dense. -C denotes the dataset subset with only compositional relations. For both subsets, the addition of LEAF significantly increases F1 score. Although the performance marked with † is better than BERT$_{BASE}$ + LEAF, the overall performance of the corresponding baseline is lower than BERT$_{BASE}$ + LEAF 1.4 and 3.3 F1 on MATRES and TB-Dense.

knowledge and generalize to downstream tasks.

Ablation study for pre-training data. To verify the value of both atomic and compositional relations, we pre-train BERT$_{BASE}$ without atomic and compositional relations acquired by LEAF. Results can be found in the Table 3. We find that for both datasets, removing either atomic or compositional relations in the pre-training stage leads to worse performance than training with both relations. Especially, we find that although there are only 140K acquired compositional relations, training with these 140K relations performs on par with training with 3.8M atomic relations. This further emphasizes the contribution of considering compositional relations to temporal relation tasks.

4.4 Predicting Compositional Relations

It is intuitive that correctly extracting compositional relations is more challenging than identifying atomic relations. In this section, we explore the capability of our model to extract challenging compositional relations. We take the subset of MATRES and TB-Dense that contain compositional relations, and evaluate both base models BERT$_{BASE}$ and RoBERTa$_{BASE}$ and their LEAF-enhanced counterparts. Results are displayed in Table 4. We observe that further pre-training with the relation supervision derived from LEAF enhances base models at identifying compositional relations. This is likely due to us giving explicit compositional relation supervision during pre-training.

We also perform ablation studies to evaluate the impact of atomic and composition relations acquired by LEAF on the subsets MATRES-C and TB-Dense-C. As shown in Table 4, we find that BERT$_{BASE}$ trained with only compositional relations performs better on both datasets than the model trained with only atomic relations. It even surpasses BERT$_{BASE}$ + LEAF, which is trained with the whole set of acquired relation supervision. This verifies that the compositional relations extracted are effective at assisting the model in tackling challenging compositional relation extraction. However, as shown in Table 3, training with the mere compositional relations does not bring better overall performance on MATRES and TB-Dense. Overall, training with all the relations obtained by LEAF is a better solution that achieves competitive overall extraction performance and predicts challenging temporal relations with greater accuracy.

4.5 Case Study

In this section, we examine specific instances where the model demonstrates an ability to grasp patterns that were not seen during pre-training as well as instances which display the model's capacity to effectively learn atomic and compositional patterns.

We present cases that confirm the effectiveness of exposing the model to out extracted patterns patterns during pre-training. In Figure 3, we observe examples where the model learns to correctly identify atomic and compositional relations after pre-training. These are examples in which the model fails to identify the relationship correctly without pre-training, and succeeds after pre-training. This shows the effectiveness of our patterns, equipping the model with a robust understanding of complex relations, enhancing its ability to make accurate

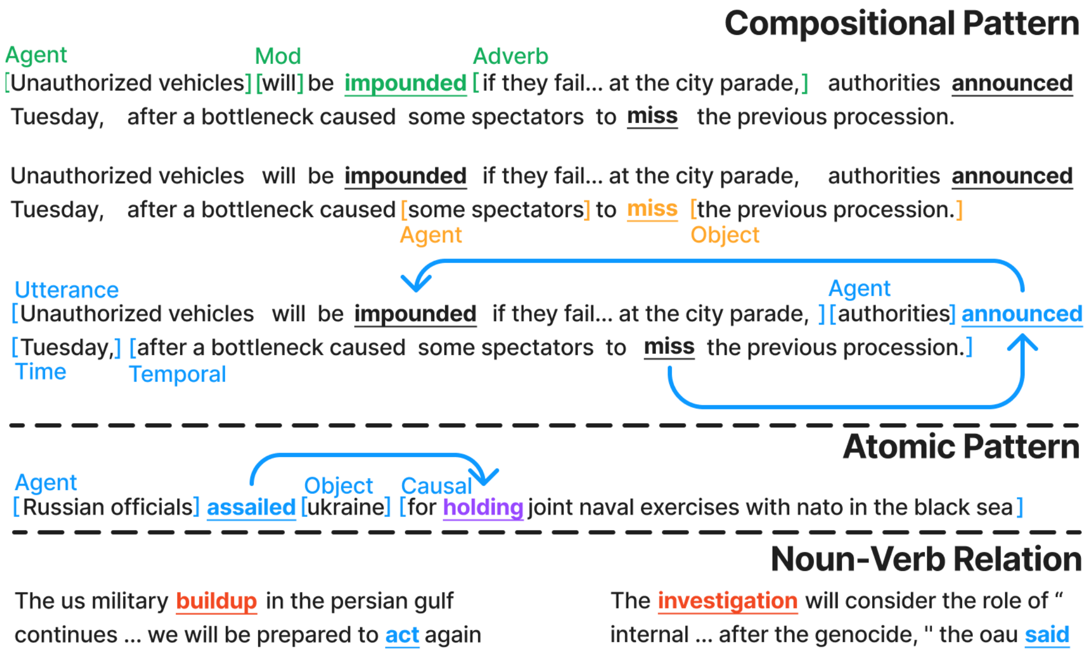

Figure 3: Examples of relations that were learned by pre-training with acquired patterns. These are instances in which the model labels the relation incorrectly without pre-training, and correctly after pre-training. For the compositional pattern, we see the event "announced" revealing a relation between "impounded" and "miss" (sentence 3). These otherwise do not have a trivial relation, because "miss" is not in the arguments of "impounded" (sentence 1) and "impounded" is not in the arguments of "miss" (sentence 2). Our model effectively identifies this relation after pre-training. For the atomic pattern, we see that vanilla RoBERTa fails on atomic relations, while LEAF can help to capture this case. For the noun-verb relation sentences, we see two examples where the model learns relations between a noun event (in red) and a verb event (in blue), despite not having seen any during pre-training.

predictions and adapt to new, unseen data.

Because the SRL parser only annotates verb events during pre-training, our model only sees verb-verb relations during pre-training. Despite this fact, LEAF has shown a remarkable ability to learn noun-verb relations that are not acquired without the pattern supervision. This is evidenced by the two examples illustrated in Figure 3. The model's ability to grasp these relations suggests that the patterns we provided during training have a potential beyond their explicit supervision.

5 Conclusions

In conclusion, our proposed LEAF framework demonstrates the effectiveness of using diverse linguistic structures to extract rich temporal knowledge of events from large-scale corpora. The extracted knowledge is able to enhance language models via a simple pre-training procedure. Our approach outperforms or rivals previous models on MATRES and TB-Dense, and excels at identifying complex compositional event relations.

6 Limitations

Our model's scope for event relations does not include all types of events. Specifically, the captured temporal relationships used for pre-training supervision does not cover noun-verb and noun-noun event pairs. Another limitation is that our model is only as good as the SRL annotations are. If the SRL annotations are noisy, then so will be our data. Also, due to the limits of computation resources, the scale of our base models are only around 110M parameters. We hope to extend to larger-scale experiments once better computational resources are available for use.

Acknowledgement

We would like to thank the anonymous reviewers for the insightful feedback and constructive suggestions. We also greatly appreciate the help of UCLA undergraduate students Eric Lin and Eric Wu in verifying the accuracy of our collected relations. This research is partially supported by an ONR grant #N00014-23-1-2780 and a Cisco Sponsored Research Award.

References

Miguel Ballesteros, Rishita Anubhai, Shuai Wang, Nima Pourdamghani, Yogarshi Vyas, Jie Ma, Parminder Bhatia, Kathleen McKeown, and Yaser Al-Onaizan. 2020. Severing the edge between before and after: Neural architectures for temporal ordering of events. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5412–5417, Online. Association for Computational Linguistics.

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, Baltimore, Maryland. 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.

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional LSTM over dependency paths. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–6, Vancouver, Canada. Association for Computational Linguistics.

Yao Cheng, Peter Anick, Pengyu Hong, and Nianwen Xue. 2013. Temporal relation discovery between events and temporal expressions identified in clinical narrative. *Journal of Biomedical Informatics*, 46:S48–S53. Supplement: 2012 i2b2 NLP Challenge on Temporal Relations in Clinical Data.

Timothy Chklovski and Patrick Pantel. 2004. VerbOcean: Mining the web for fine-grained semantic verb relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 33–40, Barcelona, Spain. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jennifer D'Souza and Vincent Ng. 2013. Classifying temporal relations with rich linguistic knowledge. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 918–927, Atlanta, Georgia. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2019. Embedding time expressions for deep temporal ordering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4400–4406, Florence, Italy. Association for Computational Linguistics.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

Rujun Han, I-Hung Hsu, Jiao Sun, Julia Baylon, Qiang Ning, Dan Roth, and Nanyun Peng. 2021a. ESTER: A machine reading comprehension dataset for reasoning about event semantic relations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7543–7559, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rujun Han, Xiang Ren, and Nanyun Peng. 2021b. ECONET: Effective continual pretraining of language models for event temporal reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5367–5380, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Quzhe Huang, Yutong Hu, Shengqi Zhu, Yansong Feng, Chang Liu, and Dongyan Zhao. 2023. More than classification: A unified framework for event temporal relation extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9631–9646, Toronto, Canada. Association for Computational Linguistics.

EunJeong Hwang, Jay-Yoon Lee, Tianyi Yang, Dhruvesh Patel, Dongxu Zhang, and Andrew McCallum. 2022. Event-event relation extraction using probabilistic box embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 235–244, Dublin, Ireland. Association for Computational Linguistics.

Carina Kauf, Anna A. Ivanova, Giulia Rambelli, Emmanuele Chersoni, Jingyuan S. She, Zawad Chowdhury, Evelina Fedorenko, and Alessandro Lenci. 2022. Event knowledge in large language models: the gap between the impossible and the unlikely.

Mayuko Kimura, Lis Kanashiro Pereira, and Ichiro Kobayashi. 2022. Effective masked language modeling for temporal commonsense reasoning. In *2022*

Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS), pages 1–4.

Hee-Jin Lee, Yaoyun Zhang, Min Jiang, Jun Xu, Cui Tao, and Hua Xu. 2018. Identifying direct temporal relations between time and events from clinical notes. *BMC Medical Informatics and Decision Making*, 18(Suppl 2):49.

Qian Li, Jianxin Li, Jiawei Sheng, Shiyao Cui, Jia Wu, Yiming Hei, Hao Peng, Shu Guo, Lihong Wang, Amin Beheshti, and Philip S. Yu. 2022. A survey on deep learning event extraction: Approaches and applications.

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.

Junru Lu, Xingwei Tan, Gabriele Pergola, Lin Gui, and Yulan He. 2022. Event-centric question answering via contrastive learning and invertible event transformation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2377–2389, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mingyu Derek Ma, Jiao Sun, Mu Yang, Kung-Hsiang Huang, Nuan Wen, Shikhar Singh, Rujun Han, and Nanyun Peng. 2021. EventPlus: A temporal event understanding pipeline. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 56–65, Online. Association for Computational Linguistics.

Inderjeet Mani, Marc Verhagen, Ben Wellner, Chong Min Lee, and James Pustejovsky. 2006. Machine learning of temporal relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 753–760, Sydney, Australia. Association for Computational Linguistics.

Paramita Mirza. 2014. Extracting temporal and causal relations between events. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 10–17, Baltimore, Maryland, USA. Association for Computational Linguistics.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037, Copenhagen, Denmark. Association for Computational Linguistics.

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

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

OpenAI. 2023. Chatgpt. `https://openai.com/blog/chatgpt/`. Accessed on May 3, 2023.

Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 13–16, Suntec, Singapore. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Peng Shi and Jimmy Lin. 2019. Simple BERT models for relation extraction and semantic role labeling. *CoRR*, abs/1904.05255.

Fiona Anting Tan, Ali Hürriyetoğlu, Tommaso Caselli, Nelleke Oostdijk, Tadashi Nomoto, Hansi Hettiarachchi, Iqra Ameer, Onur Uca, Farhana Ferdousi Liza, and Tiancheng Hu. 2022. The causal news corpus: Annotating causal relations in event sentences from news. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2298–2310, Marseille, France. European Language Resources Association.

Xingwei Tan, Gabriele Pergola, and Yulan He. 2021. Extracting event temporal relations via hyperbolic geometry. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8065–8077, Online and Punta Cana, Dominican Republic. 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.

Meiguo Wang, Benjamin Yao, Bin Guo, Xiaohu Liu, Yu Zhang, Tuan-Hung Pham, and Chenlei Guo. 2022a. Joint goal segmentation and goal success prediction on multi-domain conversations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 505–509, Gyeongju, Republic of Korea. International Committee on 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. 2022b. 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.

Haoyang Wen and Heng Ji. 2021. Utilizing relative event time to enhance event-event temporal relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10431–10437, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

QunLi Xie, JunLan Pan, Tao Liu, BeiBei Qian, XianChuan Wang, and Xianchao Wang. 2022. A survey of event relation extraction. In *Frontier Computing*, pages 1818–1827, Singapore. Springer Nature Singapore.

Chenhan Yuan, Qianqian Xie, and Sophia Ananiadou. 2023. Zero-shot temporal relation extraction with chatgpt.

Chong Zhang, Jiagao Lyu, and Ke Xu. 2023. A storytree-based model for inter-document causal relation extraction from news articles. *Knowledge and Information Systems*, 65:827–853.

Shuaicheng Zhang, Qiang Ning, and Lifu Huang. 2022. Extracting temporal event relation with syntax-guided graph transformer. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 379–390, Seattle, United States. Association for Computational Linguistics.

Xinyu Zhao, Shih-Ting Lin, and Greg Durrett. 2021. Effective distant supervision for temporal relation extraction. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 195–203, Kyiv, Ukraine. Association for Computational Linguistics.

Ben Zhou, Qiang Ning, Daniel Khashabi, and Dan Roth. 2020a. Temporal common sense acquisition with minimal supervision. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7579–7589, Online. Association for Computational Linguistics.

Guangyu Zhou, Muhao Chen, Chelsea J T Ju, Zheng Wang, Jyun-Yu Jiang, and Wei Wang. 2020b. Mutation effect estimation on protein–protein interactions using deep contextualized representation learning. *NAR Genomics and Bioinformatics*, 2(2). Lqaa015.

Appendix

A List of Atomic Patterns

In Table 5, we provide a comprehensive list of patterns used to extract the data. The top section outlines general semantic tag patterns. If a target event possesses any of these arguments, all argument events will hold the specified temporal relationship with the target event. The bottom section includes tag and beginning word patterns, consisting of a three-letter capitalized tag followed by a word. If an argument begins with such a keyword, all events within the argument will hold the temporal relation with the target event. The `to` pattern specifies that all semantic arguments beginning with `to` indicate that the target event occurs before the events in the tag. `Modal verbs` indicates that any argumentative event modified by a modal verb will hold the designated temporal relationship with the target event.

B Dataset Statistics

In Table 6, we display the statistics for both datasets. Both datasets provide gold event labels, and the task is to predict the temporal relation between two provided events.

C ChatGPT Prompts

Below are the three ChatGPT prompts that we averaged the performance over. For three-shot, we simply repeated the prompt four times, with the first three times also including the answer to the passage. Note that all examples are the ones we used for MATRES. For TB-Dense, because there are more labels, we added more options for ChatGPT to choose from. For each example, the example sentence replaces {sentence}, and the names of the left and right event replace {left_event} and {right_event}.

1. Context: {sentence}
   Based on the above paragraph, what can we conclude about the events "{left_event}" and "{right_event}"?
   Please choose one of the following:
   - "{left_event}" started before "{right_event}"
   - "{left_event}" started after "{right_event}"
   - "{left_event}" and "{right_event}" started simultaneously
   - The temporal relationship between "{left_event}" and "{right_event}" is vague

2. Read the following and determine the temporal relationship between the events "{left_event}" and "{right_event}":
   Context: {sentence}
   Options:
   - "{left_event}" started before "{right_event}"
   - "{left_event}" started after "{right_event}"
   - "{left_event}" and "{right_event}" started simultaneously
   - The temporal relationship between "{left_event}" and "{right_event}" is vague

3. Description: Given a passage, and two events "{left_event}" and "{right_event}", determine the temporal relationship between the events, choosing between one of the following options:
   - "{left_event}" started before "{right_event}"
   - "{left_event}" started after "{right_event}"
   - "{left_event}" and "{right_event}" started simultaneously
   - The temporal relationship between "{left_event}" and "{right_event}" is vague
   Passage: {sentence}

Names	Temporal Relations	Example Sentences	Explanations
CAU	After	John cooked noodles [because he was hungry].	John cooked after he was hungry
PRP	Before	John cooked noodles [for the family gathering].	John cooked before the family gathering
PPT	After	John cooked [freshly made noodles].	The noodles were made before John cooked them
to	Before	John cooked noodles [to cure his boredom]	John cooked before his boredom was cured
TMP when	After	[When he got hungry], John cooked noodles.	John cooked after he got hungry
TMP following...	After	John cooked noodles [following a request from Adam].	John cooked after Adam requested
TMP after...	After	John cooked noodles [after Adam arrived].	John cooked after Adam arrived
TMP before...	Before	John cooked noodles [before Adam arrived].	John cooked before Adam arrived
TMP during...	Simultaneous	John cooked noodles[during the storm].	It stormed while John cooked noodles
TMP while...	Simultaneous	John cooked noodles [while it was snowing].	It snowed while John cooked noodles
TMP as...	Simultaneous	John cooked noodles [as Adam arrived].	Adam arrived while John cooked noodles
ADV while...	Simultaneous	John cooked noodles, [while Adam was unamused by his jokes].	Adam was unamused while John cooked noodles
ADV if...	After	John cooks noodles [if he is bored].	John cooks after he is bored
Modal Verbs	Before	John cooked noodles and [Adam will eat them].	John cooks noodles before Adam eats

Table 5: Full list of atomic patterns. Three letter abbreviations indicate semantic tags. Patterns that only consist of a tag (e.g., PPT) indicate that all events in that tag hold the respective temporal relation to the target verb. The patterns that have a tag (e.g., TMP) and a word (e.g., during) indicate the pattern whose semantic tag starts with the word.

	Train	Validation	Test	Labels
MATRES	5,036	1,296	827	Vague, before, after, simultaneous
TB-Dense	4,032	629	1,427	Vague, before, after, simultaneous, includes, is_included

Table 6: Statistics for both datasets. Note that TB-Dense has all of the labels of MATRES, plus two additional labels: includes and is_included.