Discourse-Aware In-Context Learning for Temporal Expression Normalization

Akash Kumar Gautam[1,3*] **Lukas Lange**[1] **Jannik Strötgen**[2]
[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]Karlsruhe University of Applied Sciences, Germany
[3] Saarland University, Saarland Informatics Campus, Germany
akga00001@stud.uni-saarland.de
Lukas.Lange@de.bosch.com

Abstract

Temporal expression (TE) normalization is a well-studied problem. However, the predominately used rule-based systems are highly restricted to specific settings, and upcoming machine learning approaches suffer from a lack of labeled data. In this work, we explore the feasibility of proprietary and open-source large language models (LLMs) for TE normalization using in-context learning to inject task, document, and example information into the model. We explore various sample selection strategies to retrieve the most relevant set of examples. By using a window-based prompt design approach, we can perform TE normalization across sentences, while leveraging the LLM knowledge without training the model. Our experiments show competitive results to models designed for this task. In particular, our method achieves large performance improvements for non-standard settings by dynamically including relevant examples during inference.

1 Introduction

Temporal tagging is a challenging problem for building information extraction pipelines. Traditionally, it involves first the identification of temporal expressions (TEs) from text (**extraction**), followed by a mapping to a well-defined format such as TimeML (**normalization**). Previously, approaches to dealing with this problem involved curating handwritten rules (Chang and Manning, 2012; Strötgen and Gertz, 2013) often limiting their applicability to new domains and new languages.

More recently, deep neural networks were trained for many tasks across domains and languages (Rahimi et al., 2019; Artetxe and Schwenk, 2019). However, they require an increasing amount of training data. In contrast, the advent of large language models (LLMs) (Brown et al., 2020; Rae et al., 2022) led to strong zero- and few-shot capabilities by transferring knowledge for specific downstream NLP tasks like named entity recognition, question-answering, or sequence classification. Therefore, making use of a recent LLM without training is a compelling strategy to deal with data scarcity in multilingual setups and also to diversify utility across multiple domains.

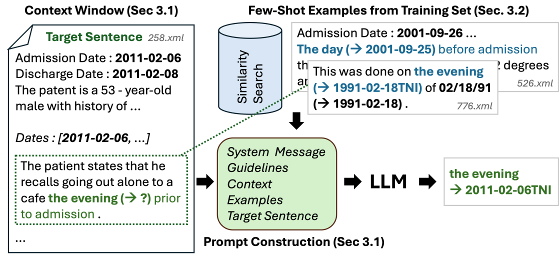

Figure 1: Overview of our proposed in-context learning approach for temporal expression normalization. Given a test input, we retrieve similar text representations from the train set. We combine both of them along with a running context window of previous predictions and feed it to a language model along with instructions.

In this work, we explore the proprietary GPT-3.5-turbo model as well as the open-source Zephyr model (Tunstall et al., 2023) for TE normalization. For both models, our discourse-aware approach (see Figure 1) leverages in-context learning using few-shot examples and a document-level temporal context window. We explore various sample selection strategies for prompting tailored toward the TE normalization task and show that standard sentence-level processing might not be suitable to capture all the necessary long-range context dependencies and discourse information. Our broad evaluation across six domains and seven languages demonstrates the competitiveness of our method to dedicated normalization models. In particular, our analysis reveals the benefits of our method in settings when the target document is more distant from their training data.

[*] Work done while the author was an intern at Bosch Center for Artificial Intelligence, Renningen, Germany

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers), pages 306–315

2 Background and Related Work

Temporal Tagging is a two-step process consisting of TE extraction from textual documents, followed by the normalization into a standard format. We follow the TimeML annotation guidelines (Pustejovsky et al., 2005), which define four temporal types, namely DATE, TIME, DURATION, and SET. For the normalization, we focus on the VALUE attribute that captures the most important temporal semantics of a TE. While explicit TEs include all necessary information for the normalization, e.g., “May 24, 2024”, others require further knowledge or temporal discourse information. For example, implicit expressions like “Easter 2024” require semantic knowledge, and relative expressions like “tomorrow” rely on an anchoring date, e.g., the Document Creation Time (DCT). Under-specified expressions are missing the relation to the anchor and cannot be fully normalized by the given context.[1]

TE Extraction and Reasoning. TE extraction has been handled as a sequence labeling problem through trained language model sequence taggers (Laparra et al., 2018; Lange et al., 2020). Lin et al. (2019) utilize BERT to identify temporal relations in text. Chu et al. (2023) investigate temporal reasoning capabilities of recent LLMs. In extraction-related tasks, prior works explore GPT’s abilities for event extraction, specifically relation extraction (Tang et al., 2023; Gao et al., 2023; Wei et al., 2023). However, no prior work studies the feasibility of TE normalization using LLMs.

For solving TE normalization, several rule-based systems have been proposed such as HeidelTime (Strötgen and Gertz, 2013) and SUTime (Chang and Manning, 2012), while other systems rely on context-free grammars (Bethard, 2013; Lee et al., 2014). However, both approaches rely on highly language-specific resources. In contrast, deep-learning-based models have demonstrated robust and generalizable performance across languages for the normalization (Lange et al., 2023). However, this system required a careful design of the neural network and large-scale training. Instead, we rely on the transfer learning abilities of LLMs by providing selected examples to learn from.

Few-Shot Learning. With the advent of powerful pre-trained language models, Brown et al. (2020) discovered that these models can be utilized for solving tasks without task-specific training. By providing examples and task descriptions, these models can generalize their existing knowledge and transfer this to follow the given instructions. Common approaches involve passing representative examples from the training set, through manual or automatic selection strategies, in task-specific formats to the LLM (Min et al., 2022; Rubin et al., 2022). Successful approaches are based on paraphrasing methods where initial text seed prompts are paraphrased into semantically similar expressions, with a further combination involving criteria like Maximal Marginal Relevance (Mao et al., 2020). As the context length of LLMs is limited and commercial APIs charge per input token, the selection criteria for sample selection becomes a crucial factor for the performance and applicability.

[1]For further details, we refer to (Strötgen and Gertz, 2016).

3 Approach

In-context learning (ICL) utilizes few-shot examples to learn the downstream task (Min et al., 2022). We follow this approach and describe our selection strategies along with prompting formats relevant to TE normalization.

3.1 Prompt Format

We follow the best practices for LLM decoding and regulating the output behavior by defining various prompt inputs. Sample prompts are given in Appendix C that showcase our prompt structure. In general, we provide information on the task, the document context, a selected set of samples, and the expected JSON output format.

For the context, we process all sentences from a document d containing temporal expressions (target sentences) sequentially, as later temporal expressions might need earlier seen temporal expressions as reference times. For this, we maintain a running record of previously seen TEs from d to support the anchoring of non-explicit expressions to the correct temporal scope, including the DCT. These running records of previously seen TEs serve as temporal containers that will allow the model to have enough semantic information to normalize relative or under-specific expressions correctly(Strötgen and Gertz, 2016).

Given a target sentence t containing one or more TEs, we aim to provide similar sentences with TEs from the candidate pool as few-shot examples. These examples are retrieved from the respective training corpora and should enable the LLM in-context learning to normalize the TEs in t.

Domain	**Ancient-Times** Wikipedia	**ECHR** Court	**ECJ** Court	**USC** Court	**i2b2** Clinical	**TempEval3** News
MLM (Lange et al., 2023)	77.0	98.2	93.5	86.8	48.1	79.0
GPT3.5 + Expert Prompt	16.5 (15)	53.2 (15)	40.5 (15)	27.3 (15)	31.3 (15)	18.2 (15)
GPT3.5 + Target-agnostic	45.3 (15)	96.0 (15)	93.1 (15)	90.4 (15)	68.3 (15)	63.6 (15)
GPT3.5 + Target-centric (Sent.)	58.1 (15)	96.3 (15)	83.2 (15)	78.4 (15)	73.6 (15)	69.9 (15)
GPT3.5 + Target-centric (Doc.)	**70.2** (5)	95.4 (5)	87.4 (1)	84.6 (1)	74.3 (15)	60.3 (15)
GPT3.5 + Target-centric + CW	63.4 (15)	**96.6** (15)	**94.2** (15)	**92.4** (15)	**76.4** (15)	**72.6** (15)
Zephyr-7B + Target-centric + CW	42.1 (5)	80.0 (15)	53.4 (1)	58.4 (10)	43.9 (10)	48.1 (15)

Table 1: TE normalization accuracy for English domains. The second number denotes the number of examples (sentences or documents) after which no performance increment was observed or the input length was exceeded. The best results using our proposed approach are in bold.

3.2 Few-Shot Example Selection

We now describe our selection strategies for the few-shot examples. For this, we use semantic search to select samples from the training sets given a target sentence t. In all setups, we use the embedding model[2] to create vector representations of text sequences and select examples based on the embedding similarity between candidate sentences and t.

Target-agnostic. The k most dissimilar examples from the training set are selected, as random sampling can lead to clusters of similar sentences. With this, we want to create a diverse and representative set that is useful for all target sentences.

Target-centric. The k most similar sentences or documents are selected given the target sentence or document. Selecting entire documents might allow the model to better learn long-term dependencies.

Target-centric + Context Window. As the LLM input length is limited, we restrict the normalization to a single sentence at a time. This allows to increase the number of selected samples without compromising the performance. To capture long-term temporal dependencies for TEs, we record previously processed sentences of the same document as a fixed-length context window (see Section 3.1).[3]

Expert Prompt. We experiment with examples derived from the TimeML guidelines. We assume that these are representative enough for the model to understand the task and the normalization format. The full prompt is given in Appendix C.

[2]https://huggingface.co/intfloat/multilingual-e5-base

[3]We use the predicted VALUE attributes as context.

4 Experiments

This section describes our experimental setup, the results, and broad analyses of various settings.

Data. For our experiments, we use 4 English datasets from various domains to evaluate the generalizability of our approach. This includes the popular TempEval3 (UzZaman et al., 2013) (news style), i2b2 (Sun et al., 2013) (clinical), Ancient-Times (Strötgen et al., 2014) (historical text) and TempCourt (Navas-Loro et al., 2019) (court decisions) datasets. The latter can be split into three subdomains, depending on the document's origin.[4] We study multilingual in-context learning with AncientTimes resources from six languages: Arabic, Dutch, French, German, Spanish, and Vietnamese. We report average accuracy across 3 different runs as the evaluation metric for all normalization experiments and use the TempEval3 evaluation script for temporal tagging setups.

Models. We experiment with the proprietary GPT-3.5-turbo[5] and the open-source Zephyr model,[6] which is considered the best-performing open-source 7B-parameter model at the time of writing [7]. The maximum input lengths are 16K and 4K tokens, respectively.

Since we model TE normalization as a text completion task, we set the temperature parameter to 0 to reduce randomness in the results. All other parameters are kept at their default values. The final input consists of four distinct types of prompts as described in Section 3.1. The context window

[4]European Court of Justice (ECJ), United States Supreme Court (USC), European Court of Human Rights (ECHR).

[5]https://platform.openai.com/docs/models/gpt-3-5

[6]https://huggingface.co/HuggingFaceH4/zephyr-7b-beta

[7]We provide results with other language models in Appendix A

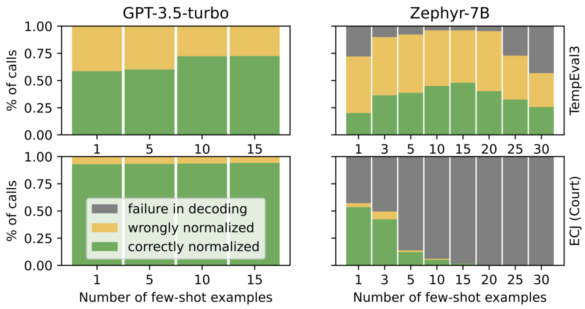

Figure 2: Analysis of how the number of examples influences the correctness and failures, e.g., for when the examples exceed the limited context length.

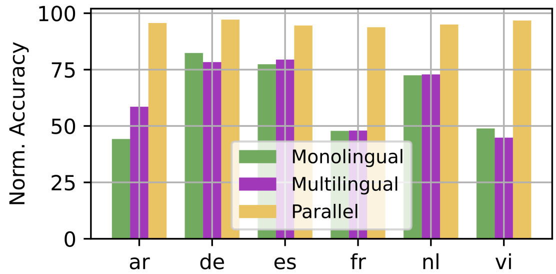

Figure 3: Performance on multilingual Ancient-Times corpora with three different sample selection pools.

length is set to 3, as it showed the highest performance in our initial experiments.

4.1 Results

The results of our different sample selection strategies are provided in Table 4. The necessity of thoroughly selected few-shot examples is emphasized across all datasets, as these methods outperform the expert prompt by a large margin. In particular, the Target-centric + Context Window approach delivers the best ICL performance for five out of six datasets. All of these datasets have a large share of explicit expressions that benefit more from additional examples than document-length context. In contrast, the narrative AncientTimes has dependencies between TEs that can be effectively dealt with only when entire documents are used. This emphasizes that the ICL method should be chosen according to the documents' characteristics.

The GPT model achieves comparable results to the MLM baseline (Lange et al., 2023), except for the clinical i2b2 corpus. In this setting, the target dataset is most distant from the training data of the MLM model, whereas our ICL methods benefit from domain-specific examples. The GPT model also considerably outperforms the smaller open-source Zephyr model, which only achieves good performance on the simplest ECHR dataset. Nonetheless, this shows the prospects of ICL for complex tasks and also for smaller models.

4.2 Analysis

We now study different aspects of our method in more detail, i.e., the effects of varying number of few-show examples and different context window lengths. We further investigate the application of our method in multilingual setups and in temporal tagging pipelines.

Number of Few-Shot Examples. As shown in Figure 2, both LLMs reach their performance peak with 10 or 15 examples for the TempEval3 corpus. However, the Zephyr model cannot benefit from more examples for the longer ECHR documents. Here, we noticed two failure types for the Zephyr model: (1) The LLM does not output machine-readable JSON, when there are not enough examples to learn the output format. (2) The model exceeds the context lengths with an increasing number of examples. This is partly due to the model's inability to follow the instructions and learn due to limited input context length.

Multilingual In-Context Learning. To evaluate if the GPT model can generalize from multilingual examples, we study the effect of our method in 3 different settings on the multilingual AncientTimes corpus. *Monolingual:* For each language, we pick same-language samples from the training set. *Multilingual:* We choose samples across languages from the combined training sets of all languages. *Parallel:* Examples were taken from the train and the test split of all languages, except the target language. The results are given in Figure 3. The general trend suggests that multilingual samples can improve performance, while the highest gain is observed with parallel data. This emphasizes that LLMs can be used for multiple languages without creating language-specific resources, e.g., by translating existing resources.

Application in a Temporal Tagging System. We couple our method with an extraction model, i.e., the NER extraction model from (Lange et al., 2023), to perform full temporal tagging. For this, we train a domain-adapted version of the sequence tagging model to match our data sources. The results are given in Table 2 and demonstrate the applicability of our method on extractions from real systems. As a baseline, we tried to use our ICL method for the extraction such that we could utilize

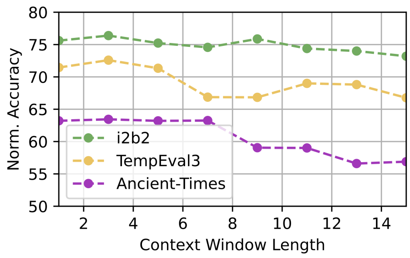

Figure 4: Effect of different context window lengths for our Target-centric + Context Window approach on 3 different corpora.

the GPT model as an end-to-end system. However, the poor recall of temporal expressions massively limits the performance of this approach, and ICL for TE extraction would have to be tackled as a separate research question.

	TempEval3 Extract.	Norm.	**ECJ (Court)** Extract.	Norm.
HeidelTime (Strötgen and Gertz)	84.1	80.0	43.3	43.0
NER+MLM (Lange et al.)	82.8	70.5	69.5	66.0
GPT-ICL (end-to-end)	52.1	40.5	24.6	18.2
Domain-NER+MLM	91.5	74.5	94.5	90.8
Domain-NER+**GPT-ICL**		81.2		91.1

Table 2: Application of different normalization methods in real-world extraction+normalization settings.

Effect of Context Window Length. Figure 4 studies context lengths for three datasets, where the best results are obtained with context lengths between one and five sentences. For longer context sizes, we observe a decrease in performance. This suggests that shorter contexts are often sufficient for LLMs to resolve temporal dependencies. Note that the AncientTimes corpus, which benefits from document-level context, does not benefit from an increased context window. We assume that the studied window size may still be too limited for the long-distance dependencies in this setting.

Error Analysis. We conduct a manual error inspection of 115 TEs from the TempEval3 corpus regarding their realizations as defined by (Strötgen and Gertz, 2016) plus vague references like “now” which is normalized to PRESENT_REF. The results are provided in Table 3 and show that our method is able to correctly normalize most explicit, vague, and implicit expressions. The latter benefit from the world knowledge in the LLM. Most challenging are relative and under-specified expressions, where the model lacks enough context or fails to incorporate context information.

	Explicit	**Implicit**	**Relative**	**Under-specified**	**Vague**
Correct	38	17	23	05	13
False	04	03	08	04	00

Table 3: Error analysis w.r.t. different realizations of TEs on examples from the TempEval3 corpus on our approach.

5 Conclusions

In this paper, we demonstrated that recent LLMs are capable of temporal expression normalization when being prompted with an appropriate in-context learning method. Our discourse-aware prompt allows the LLM to capture important context information while still being generic enough to provide general task descriptions. Our experiments across domains and languages showcase the competitive performance of our method compared to specifically designed normalization models and outperforms them when the target document is more distant from their underlying training data.

Limitations

Our experiments were limited to seven languages and our insights may not hold for untested languages. Recent advanced literature on example selection strategies presents promising avenues to impart temporal reasoning abilities for improved TE normalization in the ever-growing zoo of LLMs.

References

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Steven Bethard. 2013. A synchronous context free grammar for time normalization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, volume 2013, page 821. NIH Public Access.

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

Angel X Chang and Christopher D Manning. 2012. Sutime: A library for recognizing and normalizing time expressions. In *Lrec*, volume 12, pages 3735–3740.

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

Jun Gao, Huan Zhao, Changlong Yu, and Ruifeng Xu. 2023. Exploring the feasibility of chatgpt for event extraction.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Lukas Lange, Anastasiia Iurshina, Heike Adel, and Jannik Strötgen. 2020. Adversarial alignment of multilingual models for extracting temporal expressions from text. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 103–109.

Lukas Lange, Jannik Strötgen, Heike Adel, and Dietrich Klakow. 2023. Multilingual normalization of temporal expressions with masked language models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1174–1186.

Egoitz Laparra, Dongfang Xu, and Steven Bethard. 2018. From characters to time intervals: New paradigms for evaluation and neural parsing of time normalizations. *Transactions of the Association for Computational Linguistics*, 6:343–356.

Kenton Lee, Yoav Artzi, Jesse Dodge, and Luke Zettlemoyer. 2014. Context-dependent semantic parsing for time expressions. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1437–1447.

Chen Lin, Timothy Miller, Dmitriy Dligach, Steven Bethard, and Guergana Savova. 2019. A bert-based universal model for both within-and cross-sentence clinical temporal relation extraction. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 65–71.

Yuning Mao, Yanru Qu, Yiqing Xie, Xiang Ren, and Jiawei Han. 2020. Multi-document summarization with maximal marginal relevance-guided reinforcement learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1737–1751.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022. Metaicl: Learning to learn in context. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2791–2809.

María Navas-Loro, Erwin Filtz, Víctor Rodríguez-Doncel, Axel Polleres, and Sabrina Kirrane. 2019. Tempcourt: evaluation of temporal taggers on a new corpus of court decisions. *The Knowledge Engineering Review*, 34:e24.

James Pustejovsky, Robert Ingria, Roser Sauri, José M Castaño, Jessica Littman, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Inderjeet Mani. 2005. The specification language timeml.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2022. Scaling language models: Methods, analysis & insights from training gopher.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for ner. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671.

Jannik Strötgen, Thomas Bögel, Julian Zell, Ayser Armiti, Tran Van Canh, and Michael Gertz. 2014. Extending HeidelTime for temporal expressions referring to historic dates. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2390–2397, Reykjavik, Iceland. European Language Resources Association (ELRA).

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47:269–298.

Jannik Strötgen and Michael Gertz. 2016. Domain-sensitive temporal tagging. In *Domain-Sensitive Temporal Tagging*, pages 47–83. Springer.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining?

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA. Association for Computational Linguistics.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-shot information extraction via chatting with chatgpt.

A Further Analysis

In this section, we share the results of other 7B-parameter language models that we explored for the task of TE normalization. We present the results with Llama2 (Touvron et al., 2023), Mistral (Jiang et al., 2024), NeuralTrix [8], and Westlake [9]. We observe a large performance gap for all of these models in comparison to the Zephyr model. Upon manual error inspection, we found severe problems regarding their ability to follow instructions, and therefore, to produce valid json outputs.

B Implementation Details

We used Faiss[10] to index and cluster vector representations of text sequences (sentences or documents) throughout this work. Spacy[11] was used to split into sentences. The dissimilarity threshold value for the Target-agnostic approach was set to 0.7. For GPT-3.5-turbo, we also ensure that `system_fingerprint` field was consistent across all experiments for the online API calls.[12]

C Detailed Prompt Information

To enable the dynamic and conversational abilities of GPT-3.5-turbo, we make use of `messages`[13] that include further information on how the output and response should be produced. These are intended to pass enough context for the conversation model to understand the nuances of the task.

Figure 5 includes a prompt example passed for Target-centric + CW approach for a sample document from the test set.

We now describe the different types of prompt components, that make up the final API call.

`SYSTEM PROMPT`: Used to provide system-level instructions to guide the model's behavior throughout the conversation.

`USER PROMPT`: Used to specify the user role for the text input.

`ASSISTANT PROMPT`: Instructions on how the model should respond to the user-level instructions.

`GUIDELINES PROMPT`: Consists of actual text sequences in TimeML annotation format from the train (few-shot examples) and test (target sentence) splits.

C.1 Expert Prompt

Figure 6 includes the text sequences that were passed as guidelines prompt for the expert prompt example selection strategy mentioned in Section 3.1.

[8] https://huggingface.co/CultriX/NeuralTrix-7B-dpo

[9] https://huggingface.co/senseable/Westlake-7B

[10] https://github.com/facebookresearch/faiss

[11] https://github.com/explosion/spaCy

[12] https://platform.openai.com/docs/guides/text-generation/reproducible-outputs

[13] https://community.openai.com/t/on-using-the-messages-array-with-gpt-3-5-turbo-and-gpt-4/367376

Domain	Ancient-Times Wikipedia	ECHR Court	ECJ Court	USC Court	i2b2 Clinical	TempEval3 News
GPT3.5	63.4 (15)	**96.6** (15)	**94.2** (15)	**92.4** (15)	**76.4** (15)	**72.6** (15)
Zephyr-7B	42.1 (5)	80.0 (15)	53.4 (1)	58.4 (10)	43.9 (10)	48.1 (15)
Llama2-7B	8.0 (5)	16.7 (15)	5.1 (5)	41.1 (15)	2.6 (10)	7.75 (10)
Mistral-7B	6.8 (5)	27.3 (15)	2.0 (1)	35.3 (5)	1.6 (5)	7.0 (10)
NeuralTrix	4.7 (5)	16.7 (15)	6.7 (15)	41.1 (15)	2.5 (10)	7.8 (15)
Westlake-7B-v2	3.2 (5)	16.7 (15)	6.6 (15)	41.1 (15)	2.4 (10)	7.8 (15)

Table 4: TE normalization accuracy with other language models. The second number denotes the number of examples (sentences or documents) after which no performance increment was observed or the input length was exceeded. All LLMs were prompted with our Target-centric + CW method

SYSTEM PROMPT: Function as a system that gives the normalized time expressions for all TIMEX3 tags of type DATE, TIME, DURATION, and SET. The identified normalized time expression should be according to TIMEML annotation standards. The output shows the normalized values for the time expressions. All time expressions that are required to be normalized is passed as a list.

USER PROMPT: Are you clear about your role?

ASSISTANT PROMPT: Sure, I'm ready to help you with your task. Please provide me with the necessary information to get started.

GUIDELINES PROMPT: Here are some examples and the expected output format with normalized expressions

1. She will need to continue for at least `<TIMEX3 tid="t17" type="DURATION" previous_timex="2002-02-01 2002-02-08" dct="2002-02-01">10 more days</TIMEX3>` or as clinically indicated by the course of her cellulitis.
   List of time expressions to normalize: `['10 more days']`
   Output: `{'10 more days': 'P10D'}`

2. Sentence: The patient did well and her suprapubic tube was clamped starting on `<TIMEX3 tid="t10" type="DATE" value="1993-07-13">postoperative day four</TIMEX3>`. Clamping continued until `<TIMEX3 tid="t11" type="DATE" value="1993-07-15">postoperative day six</TIMEX3>`. By `<TIMEX3 tid="t12" type="DATE" value="1993-07-15">postoperative day six</TIMEX3>`, the patient was also tolerating a regular diet and passing flatus. In addition, she will take Ciprofloxacin for `<TIMEX3 tid="t13" type="DURATION" previous_timex="1993-07-13 1993-07-15" dct="1993-07-09">nine more days</TIMEX3>`.
   List of time expressions to normalize: `['nine more days']`
   Output:

Figure 5: Prompt Example passed to GPT-3.5 for Target-centric + CW (context window) approach. In the guidelines prompt, sentence #1 is the text sequence picked from the train set. Sentence #2 includes text sequences from the test set. Text highlighted in blue is the target sentence passed to the LLM model for normalization. Ones marked in red, are part of the running context window (previous sentences in the same document from the test set, where the VALUE attribute is replaced by predictions from the model.)

GUIDELINES PROMPT:

1. Reference for ruling visas would be given on `<TIMEX3 type="DATE" tid="t2">30 April 2013</TIMEX3>`. Written regards to further procedures would be made public on `<TIMEX3 type="DATE" tid="t3">8 May 2014</TIMEX3>`. `<TIMEX3 type="DATE" tid="t4">The following day</TIMEX3>` the house will open for discussion. The ceremony for delegates on current immigration laws are held `<TIMEX3 type="SET" tid="t5">annually</TIMEX3>`. Such kinds of meetings usually lasts only `<TIMEX3 type="TIME" tid="t6">30 minutes</TIMEX3>`. Such meetings have been going on now for `<TIMEX3 type="DURATION" tid="t7">more than five years</TIMEX3>` now. Mr. Mark filed for an extension just `<TIMEX3 type="DURATION" tid="t7">30 days</TIMEX3>` before the expiry of his credentials. `<TIMEX3 tid="t8" type="DATE">previously</TIMEX3>` he did not do it. In `<TIMEX3 type="DATE" tid="t9">2016</TIMEX3>` last such case occurred.

   List of time expressions to normalize: ['30 April 2013', '8 May 2014', 'the following day', 'annually', '30 minutes', 'more than five years', '30 days', '2016', 'previously']
   Output: {'30 April 2013': '2013-04-30', '8 May 2014': '2014-05-08', 'the following day': '2014-05-09', 'annually': 'P1Y', '30 minutes': 'PT30M', 'more than five years': 'P5Y', '30 days': 'P30D', '2016': '2016', 'previously': 'PAST_REF'}

2. Sentence:

   Output:

Figure 6: Text sequences that were passed as expert prompt example selection strategy. Sequence in Sentence #2 would be the one from the test set.