Temporal Expression Recognition in Legal Transcripts

Elizabeth J. Goldstein, Maria Berger
ORRO AI Genius, Ruhr University Bochum
USA, Germany
lj@orrogenius.com, maria.berger-a2l@rub.de

Abstract

In litigation, trial transcripts provide verbatim records of witness testimony, primarily given in response to attorney questioning. To effectively analyze these transcripts, lawyers must often reconstruct events in chronological order—a task that begins with identifying dates associated with testified facts. This paper introduces two datasets for temporal expression extraction from legal transcripts: a primary dataset derived from a lengthy 1995 U.S. criminal trial, and a smaller robustness-testing dataset drawn from seven other legal proceedings. We evaluate semi-supervised approaches for date entity recognition, fine-tuning neural models on weakly labeled training data, and benchmarking them against both small and large language models. Our best-performing models achieve 83% F1-score on the primary dataset (FLAIR rule-modified) and 72% F1-score on the cross-domain, small test set (BERT-cased). These results, alongside our annotated datasets and corresponding experiments, provide a foundation for developing robust date extraction and temporal ordering tools for speech-derived legal text. Moreover, we identify unique challenges for state-of-the-art NER models on legal transcripts, including legal terminology and multiple anchor date resolution.

Keywords: temporal expression extraction, date normalization, span identification, date annotation

1. Introduction

This work's purpose is to take the first steps in automating timelines for litigation analysis. Timelines often provide the most effective way to visualize and contextualize complex situations, such as studying historical facts and critical business decisions (Adak et al., 2022).

As legal transcripts present complex scenarios, timelines can be an important tool in examining the strength of a party's case. Timelines require two main steps. These are: (1) temporal expression (TE) extraction, defined as identifying text spans, which convey temporal information (Wu et al., 2005); and (2) temporal tagging, also known as temporal annotation, which combines extraction and normalization (Lange et al., 2023). Normalization converts temporal expressions into a standardized format, thus allowing machine readability. This paper examines these tasks, which remain open problems in natural language processing (annotation sample in Tab. 1).

Limited work has focused on effective methods for the temporal annotation of legal documents. Moreover, identifying named entity recognition (NER) spans in legal transcripts has not been explored, and temporal normalization work has not addressed a situation that frequently arises with legal transcripts: relative dates may be anchored upon something other than the document creation date. In this paper, we contribute work to address these unexplored problems:

1. We provide two new transcribed legal spoken English corpora from legal proceedings. 1,736 of these sentences[1] are gold-labeled for date expression spans. A subset of 291 sentences (containing 235 dates) were normalized at the sentence and document levels. To the best of our knowledge, we are the first to introduce evaluating date normalization abilities at the sentence and document levels and providing gold labeling for both assessments.
2. We assess the performance of the temporal taggers, SUTime, HeidelTime, and Claude Sonnet 4.5 (released Sept. 29, 2025)[2] against the new dataset of 291 sentences with gold labeled date annotations at the sentence and document levels.
3. We enhanced TE extraction through a multi-stage weak supervision approach. First, we applied the FLAIR Ontonotes NER model (FLAIR OOTB, Akbik et al. (2018) trained on the OntoNotes dataset (Weischedel et al., 2012) to automatically label dates in our corpus. We then developed rule-based labeling functions to correct and augment these predictions, creating a large weakly-labeled training dataset. Finally, we used this silver-labeled data to fine-tune BERT-Cased (Devlin et al., 2018)[3] and LEGAL-BERT (Chalkidis

[1] To create the corpus, we utilized the entire transcript. As such, we included within the corpora metadata like the case caption and index from non-testimonial parts of the transcripts.

[2] Claude Anthropic. Under: https://www.anthropic.com/claude with its system card: https://assets.anthropic.com/m/12f214efcc2f457a/original/Claude-Sonnet-4-5-System-Card.pdf.

[3] https://huggingface.co/tftransformers/bert-base-cased

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 6022–6037
11–16 May 2026.

docum. created	August 3, 1995
sentence	MS. CLARKE: You have described in your review the fact that you looked at photographs of these strips at the LAPD **in January, was it 1995?**
NER-labeled	MS. [CLARKE **S-PERSON**] : You have described in your review the fact that you looked at photographs of these strips at the [LAPD **S-ORG**] in [January, **B-DATE**] [was **I-DATE**] [it, **I-DATE**] [1995? **E-DATE**]
SUTime output	[{'timex-value': **'1995-01'**, 'start': 117, 'end': 124, 'text': 'January', 'type': 'DATE', 'value': '1995-01'} {timex-value': **'1995'**, 'start': 134, 'end': 138, 'text': '1995', 'type': 'DATE', 'value': '1995'}]
correct normali.	1995-01

Table 1: Example of legal transcript date annotation and normalization

et al., 2020)[4] models for improved date expression extraction. We then compared five approaches: FLAIR OOTB, weakly-supervised, enhanced FLAIR, the two BERT variants, and Claude Sonnet 4.5, a large, generative language model.

4. Through our experiments on two NLP tasks, TE and date annotation, we identify challenges working with speech-derived text and in particular, spoken English utilizing legal terminology and conventions.

2. Background

2.1. Temporal Information and Usage

Time enables us to record, order and measure events, processes, and actions (Tissot et al., 2015). Often, the TEs extracted must be mapped to standardized formats, referred to as *normalization* (e.g. January 1, 2024 represented as 2024-01-01), to solve a downstream task. The normalization process includes finding the absolute value of relative dates such as *tomorrow* and *three weeks from Thursday*. (Gautam et al., 2024)

2.2. Date Objects Taxonomy

This research focuses on temporal expressions consisting of 24 hours or more or measured by the unit of a day or more such as *a half a day*, herein referred to as dates. Date normalization requires an understanding of different date object types. Both Wu et al. (2005) and Kuzey et al. (2016) inspired us to divide date objects into five types. These are:

1. A *Calendar Date* is a point in time or time span that can be placed on a calendar such as March 3, 2020 or December 2024 (*normalized as 2024-12*). (Wu et al., 2005)

2. A *Duration* measures the distance between two points in time such as "three days" (*normalized as P3D*). Knowing an event's start and end time or one of these plus the time's length allows one to place an event's duration on a timeline. (Wu et al., 2005) Duration includes chronological ages. (Wu et al., 2005)

3. A *Vague* time-period provides some information concerning a calendar date but not enough to place it on a timeline (e.g., *It was on a Monday.* and *sometime in July*. (Wu et al., 2005)

4. A *Set* is a periodic interval such as *every Tuesday* or *monthly (normalized as R1M)*. (Wu et al., 2005)

5. An *Anchor Date* is a calendar date for which a time expression points and is needed to calculate the date on which the speaker is referencing. Anchor date references can be implicit, such as when a speaker refers to tomorrow, or explicit, such as a week from this Monday (Ahn et al., 2005). The anchor date for *deictic* phrases such as *last week* is the utterance's creation date while the anchor date for *anaphoric* expressions, which include abbreviated date references such as *on the 20th*, point to an anchor date previously referred to in the conversation (Ahn et al., 2005).

2.3. Legal Transcripts' Purpose and Contents

Transcription converts verbal discourse into a written format, so others not involved in the proceeding can utilize the testimony in their work. This conversion process is referred to as *entextualization* (Richardson et al., 2022). American legal transcripts, such as those included in the corpora, contain mostly witness questioning and answering, but also include other material such as lawyer objections and metadata (e.g. the case caption).

[4] https://huggingface.co/nlpaueb/legal-bert-base-uncased

3. Related Work

3.1. Temporal Expression Identification through NER Tagging

The earliest temporal extraction approaches utilized the creation of hand-crafted rules. However, these approaches are labor-intensive. Moreover, they struggle with generalization and thus are not adept at tasks focused on new subject areas (Gautam et al., 2024). To create more robust temporal extraction capabilities, including the ability to assess new target expressions, researchers moved from rule-based to machine-learning methods. These new methods transformed temporal extraction into a span identification problem otherwise known as a named entity recognition (NER) task. NER identifies and categorizes entities within text by delineating their boundaries and assigning them to predefined categories such as person, geographic location, and organization. Many established open-source NER systems include date and time information tags. English NER models utilize a labeling schema, which distinguishes span edges from other span members. Machine learning NER approaches involve a large set of manually labeled text to train the machine learning model. Earlier researchers solved NER problems with token-level sequential labeling or span-level classification methods. For the token-level approach, a model labels each token to represent its entity type. Researchers have relied upon several machine learning algorithms for this sequence-labeling task, including: Hidden Markov Model, Maximum Entropy Model, and Support Vector Machine.

Span-level classification models review all possible sentence spans and classify them by entity types. Span-based approaches generally focus on complex NER tasks such as overlapping and interlocking spans, referred to as discontinuous NER. Examples of overlapping and discontinuous temporal spans in the studied dataset are in Figure 1. Researchers have utilized several methods to identify spans, including bi-affine attention (Yu et al., 2020) and hypergraph methods (Lu and Roth, 2015).

3.2. NER Analysis in Related Topics

There is limited research on developing NER models specifically for English language text derived from speech. We are not aware of any other legal transcript dataset. We are aware of several speech-to-text open-source datasets. One is the Spoken Language Understanding Evaluation (SLUE, Shon et al. (2022)), which is comprised of European Parliament event recordings transcripts with the following date and time phrases, 762-train 260-development, and 186-test. In addition, Patil et al. (2023) examines the efficacy of utilizing BERT transformer models on a dataset of approximately, 6,900 utterances with 2,055 time entity samples, derived from human dialogues relating to multi-domain task-oriented subjects, which included calendar scheduling. Two datasets providing temporal annotations of broadcasts, are the Automatic Content Extraction (ACE) Program (Doddington et al., 2004) and the Evaluation de Systemes de Transcription enrichie d'Emissions Radiophoniques (ESTER) (Galliano et al., 2005).

Au et al. (2022) provide a legal NER dataset, consisting of securities filings, which does not include labels for temporal expressions. Au et al. (2022) found that a specialized BERT model trained on legal data performed approximately 30% better than one trained on a general English dataset.

3.3. Temporal Normalization Tools

Normalization annotation schema: TimeML (Pustejovsky et al., 2003) is a rich annotation schema designed to capture a complete range of temporal information and set it in a standardized format. It is the predominant time annotation method for English texts (Lange et al., 2023) and is utilized by the date normalization tools tested in this paper.

TIMEX3 (Derczynski et al., 2012) is part of the TimeML annotation language and addresses temporal expressions. It possesses three attributes: *type* (date and time (when), duration (how long), frequency (how often)), *val* (normalized date and time TIMEX3 value), and *mod* (i.e., approximate, start and end).

Temporal normalization tools: Two popular temporal annotation tools are SUTime (Chang and Manning, 2012) and HeidelTime (Strötgen and Gertz, 2013)[5]. Both tools exploit hand-crafted rule-based systems to derive their outputs.

SUTime: SUTime (Chang and Manning, 2012), a library for recognizing and normalizing time expressions, is part of Stanford's CoreNLP pipeline. It is a rule-based system built on top of the TokensRegex library returning TIMEX3 tags or SUTime's extensions to these tags and normalizes the temporal information extracted. See Tab. 1 for a sample SUTime output. SUTime introduces a fourth temporal type, INTERVAL. An interval denotes a time range with a start and end point such as *from June to December*. SUTime is a Java library and comes with a Python wrapper.

HeidelTime: HeidelTime (Strötgen and Gertz, 2012) is an open-source rule-based, multi-lingual, cross-domain temporal tagger. For applicable languages, HeidelTime contains resources such as

[5]We accessed HeidelTime through py-heideltime 1.0.6. and SUTime through the Python sutime library.

Figure 1: Example of NER discontinuous and overlapping span from the dataset

patterns, normalization information, and rules, for outputting TIMEX3 tags and normalizing temporal information. It was the most accurate open-source tool for temporal annotation of the English documents in the TempEval-2 and TempEval-3 challenges (Verhagen et al., 2007; UzZaman et al., 2013). In addition to the normalized date and the value type, HeidelTime provides a binary value for whether the temporal expression is an approximate value.

4. Research Methodology

4.1. Dataset Formation and Tokenization

We provide a new dataset consisting of the legal transcripts covering an approximately nine-month criminal murder trial[6], which took place in the US in 1995 (referred to as the large dataset)[7]. Other transcripts, which form the smaller dataset, are derived from seven United States legal transcripts. Specifically, these transcripts originated from: (i) five civil trials, (ii) one civil deposition, and (iii) one additional criminal trial.[8] These matters occurred between 2017 and 2023. We used NLTK's sent_tokenize library to tokenize the texts into sentences. The large dataset contains over 400,000 sentences and the smaller one consists of 67,569 sentences.

We randomly shuffled the large dataset and split it into train, validation, and test sets (see final splits in Table 2). Next, we tokenized these sentences into words via the segtok tokenizer, which the FLAIR OOTB model also uses.

To make a dataset more heavily concentrated with date expressions, we reduced the dataset size by only retaining sentences containing words or phrases that the FLAIR NER tagger identified as a date or matched those on the date keyword list manually created by us.[9] This data culling resulted in the large dataset having 28,975 tokenized sentences (23,978/training, 2,966/development, 1,600/testing, 1330/unused). From the culled large dataset, we provide three annotated subsets, a training and development set with machine-generated "silver" annotations, and a test set with manual "gold" date span annotations. The first 155 of the 1,600 test data contained 132 dates, and was manually gold labeled for date normalization at sentence and document levels.

For the smaller dataset—just used for testing—we retained the first 700 sentences of each transcript and then followed the same methodology to obtain a dataset more concentrated with date expressions, resulting in a subset of 677 sentences. Then, these sentences were randomly divided into a train/test split of 80/20. However, the train set was not used in this paper. This methodology resulted in a test set of 136 sentences with 95 date spans and 100 dates. The date spans, and dates were manually gold-labeled.

For the date tagging and normalization task, both the large dataset and smaller dataset were manually gold labeled for date normalization at the sentence and document levels, see Tab. 2.

4.2. Text Preprocessing

Text preprocessing is an important step for building machine learning models, including NER models. Speaker identities [10] were removed from the sentences through manual rules, which were mostly effective. Although some court reporters transcribed the testimony in all caps and others used standard English capitalization rules, we kept the text capitalization found in the individual court transcripts for the span identification task, except for the Legal BERT model, which was uncased, and testing the SUTime date annotation tool. The HeidelTime date annotation tool did not recognize dates with words in all caps; thus, we lower-cased the text for these experiments. In addition to the normal punctuation, court reporters use a dash or double-dash to indicate that the speaker has been interrupted mid-sentence. We left this atypical punctuation unchanged.

4.3. Annotation Guidelines and Gold Labeling

Instructions were prepared for the annotation procedure. The annotation guidelines comprise best

[6]This nine-month criminal murder trial, formally *People of the State of California v. Orenthal James Simpson*, was one of the most widely followed trials in American history and related to the death of the defendant's ex-wife and a second person.

[7]From http://walraven.org/simpson/

[8]Both datasets, all annotations, and the annotation instructions are at: https://github.com/Goldstein-Berger/Temporal-Expression-Recognition-in-Legal-Transcripts.

[9]The keywords are in the Appendix.

[10]An example of a speaker identity is "Ms. CLARKE:".

data split	large data			small data		
	#sent	#date spans	#dates	#sent	#date spans	#dates
Finetuned Bert Models						
neural TE, train	23,978	18,788	-	-	-	-
neural TE, dev	2,966	2,324	-	-	-	-
neural TE, test	1,600	1,229	-	136	95	-
Rule-Based testing only						
date annotation tools: DN	155	135	132	136	95	100
FLAIR modified: TE	1,600	1,229	-	136	95	-
FLAIR OOTB						
TE	1,600	1,229	-	136	95	-
Few-Shot gen. LLM testing only						
TE	1,600	1,229	-	136	95	-
DN	155	135	132	136	95	100

Table 2: Overview on data splits for neural and rule-based T(emporal) E(xpression extraction) and D(ate) N(ormalization) performed using existing tools; according to #sentences, #date spans, and # of dates (date objects which can be normalized) in the datasets

practices and rules for the temporal expression identification and date annotation. The instructions were created based on the purposes of the research and were revised to better address edge cases after one meeting of the two annotators to review the first 30 annotations.[11] We had one annotator gold label the 1,600 test sentences from the large dataset and two annotators for the 136 test sentences from the smaller dataset.

Since the task's purpose was to produce spans that could be utilized as the first-step in normalizing dates, certain related dates within a sentence were included as part of a single span, since these dates would need to be considered together for normalization purposes. These were intervals, estimated dates given for one event (i.e. *either Monday or Tuesday)*, and dates given in error that were then immediately corrected by the speaker. Thus, *between July the 13th and–excuse me, June the 13th and July the 3rd* was labeled as one span. In addition, words of approximation or equivocation were included as part of the span because this information would be important in comparing witness testimony.

4.4. Inter-Annotator Agreement

Both annotators had experience as practicing attorneys. For the span identification task utilizing two annotators, the F1-agreement between the two annotators was 0.69 strict and 0.92 relaxed for the annotations they completed independently. The Cohen's Kappa score was 0.68. While these values are in the typical range, they highlight the task complexity. The difference between the strict and relaxed scores indicate that while annotators often struggle to clearly distinguish between span boundaries, they have a strong agreement on the existence of a temporal expression within a sentence. For the date normalization task, the agreement scores at the sentence level were higher than at the document level (see Tab 3), which was a more complex task. Document level normalization can require resolving nested temporal coreference: a target sentence may contain a relative expression (e.g., the next day) that refers to a partial date in a second sentence, which itself must be resolved using an anchor date stated in a third sentence. Such multi-hop resolution introduces compounding opportunities for annotator disagreement.

task	F1-score strict	F1-score relaxed	Cohen's Kappa
TE	.69	.92	.68
date anno. sent.	.86	.92	.84
data anno. doc.	.74	.88	.72

Table 3: Inter-annotator agreement in T(emporal) E(xpression extraction) and date anno(tation) at the sent(ence) and doc(ument) levels

After they independently annotated the sentences, they worked together to come to a consensus on all of the labeling. The consensus labels were used for the experiments.

For assessing the neural models, we adopted the BIOES tagging system utilized by the FLAIR OOTB NER Model. BIOES identifies tokens by class type and position within the span. These positions are labeled: beginning (B), intermediate (I), ending (E), singular (S), and with non-named-entity words labeled outside (O).[12]

[11] The complete annotation guidelines will be published with the paper.

[12] For ease of processing, we reduced FLAIR's tagging

The **neural models** were tested on a strict matching basis and the precision, recall, and F1-scores were calculated via the CoNLL-2003 Shared Task method, which requires strict matching. (Sang and Meulder, 2003)[13]

4.5. Gold Labeling for SUTime, HeidelTime, Claude Sonnet 4.5 as Date Annotator

To test the date annotation tools' **date normalization** abilities, sentences were provided to the tools as a single string and the tools utilized their internal word tokenizers to tokenize the given sentence. For Claude Sonnet, all sentences from that day's transcript were given to the model to be used in the date annotation task. We normalized according to TIMEX3 the dates at: (i) the sentence level, because this is the data ingested by the tools to produce their outputs, and (ii) the document level to derive the true accuracy of the tools' normalization abilities. For both the date annotations tools and Claude Sonnet's outputs, we manually removed: (i) time expressions appended to the tools' date outputs as a means to focus solely on date expressions and (ii) general time references such as "now" and "later", which were not annotated.

On both the sentence and document levels, when parts of the date could not be determined, the gold labels signified this by an "X" (E.g., the speaker says, *You talked to your superior on the 11th.*, the normalized sentence-level gold would be *XXXX-XX-11*.) For document-level normalization, all text information was utilized to determine date information, which was only implicitly provided in the sentence. For the date annotation tools **evaluation**, all TE utterances were counted as TEs regardless of whether one could explicitly or implicitly conclude that a date was stated in error due to disfluency or speaker self-correction.[14]

> MR. GOLDBERG: And would you expect there to be some degradation on the conventional markers and those stains between **July the 13th and–excuse me, June the 13th** and July the 3rd?
> *Speaker self-correction; testdata sample*

Like the date expression extraction experiments, we derived the precision, recall, and F1-scores for the **date annotation task** on a strict matching basis. (Sang and Meulder, 2003)

5. Results

5.1. Neural & Generative LLMs for TE

We fine-tuned the FLAIR OOTB Model's output via hand-crafted rules. Specifically, for the test sets, we utilized manually curated rules to extend the date spans of the FLAIR OOTB Model's date tags by finding additional date words preceding the tagged date span. When a rule found one or more date keywords prior to an identified FLAIR-tagged date span, the head of the date span was moved to the earliest new word identified via the rules.

We tested using rule-modified FLAIR inputs to fine-tune two BERT models to complete the token-classification task.[15] We did not modify the generative large language model tested. Rather, we assessed its capabilities by providing a prompt containing examples. The prompts utilized are provided in the Appendix.

5.2. Neural and Generative Large Language Models Results

Table 4 summarizes for the date span identification task the overall results of the FLAIR and BERT models fine-tuned for the date identification task[16], and Claude Sonnet 4.5.[17] Our FLAIR rule-modified (Silver) approach was the best performing model on the large dataset, surpassing all other models in precision, recall, and F1-scores by at least 5%. For the smaller dataset, the Bert cased model had the best F1-score.

To explore the FLAIR rule-modified (Silver) approach's abilities, we examined the patterns of its errors in identifying multi-word date expressions in the larger dataset. For identifying error types, each date span with one or more token tagging errors was counted as one error. Three error types were identified: false positives, false negatives, and tokens correctly tagged but assigned to the wrong date span.

There were 56 false positives, most (39%) related to the tagging exhibit numbers as dates. Another large false positive error category (23%) re-

types from 18 to 9 tags used in addition to the date tag were the most frequent ones found in the large dataset: person name, cardinal, organization name, geo-political value, ordinal value, time value, quantity, and the law tag.

[13] Scores are measured using the seqeval tool (Nakayama, 2018) in strict mode with micro-average and the "IOBES" scheme.

[14] Disfluency/speaker self-correction occurred in 2 of the 155 large dataset test sentences.

[15] Parameter settings for the BERT models: 5 epochs, learning rate 2e-5, batch size 16.

[16] For the large dataset, the Legal BERT scores on a relaxed basis are: precision: 91%, recall: 95%, F1-score: 93%.

[17] For the large dataset, Claude Sonnet 4.5 scores on a relaxed basis are: precision: 82%, recall: 93%, and F1: 87%. For the small dataset, Claude Sonnet 4.5 scores on a relaxed basis were: precision: 79%, recall: 93%, and F1: 86%.

lated to term disambiguation. The model tagged words, which possess a temporal meaning in some contexts but were not used in that particular sense in the sentence. 9% of the false positive errors arose from identifying as temporal information numbers used for other purposes such as identifying laws or providing count or ranking information (see Fig. 2).

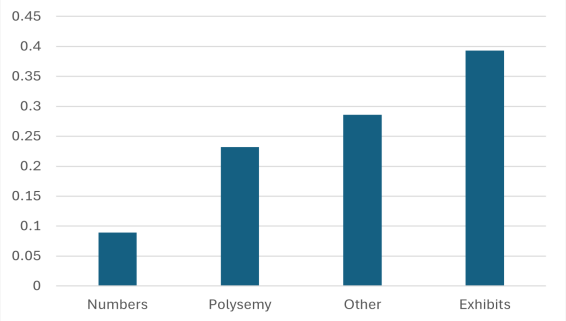

Figure 2: Fine-Tuned FLAIR OOTB Model's False Positives by Type

We conclude that the FLAIR rule-modified (Silver) approach for multi-token date spans was less successful when the English used to express the date span varied significantly in grammar and syntax from standard English, based upon measurements derived from (Sun and Wang, 2024; Crossley et al., 2023)'s model, which was built to grade English essays (see Fig. 3).

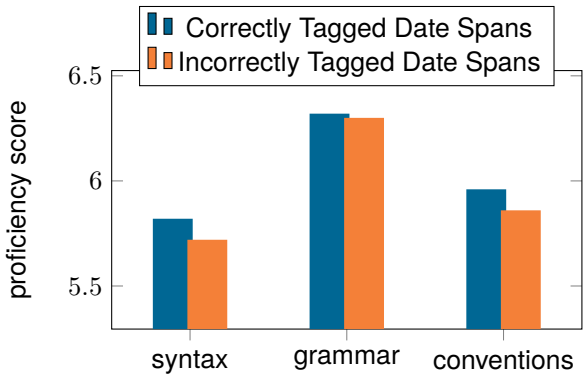

Figure 3: The FLAIR Rule-Modified (Silver) Model was better at identifying multi-token date spans when the language quality more closely resembled written English. Y-axis shows proficiency score (1-to-10) according to (Crossley et al., 2023)

Indeed, the FLAIR rule-modified (Silver) approach did not recognize as date spans some informal date patterns utilized in spoken English. These were: month-day-comma-year patterns with the year abbreviated to two digits, such as *May 10, '94*, nontemporal language bisecting a date object, such as *Sunday, which would be June the 12th* and *January, was it, 1995*. Moreover, it found challenging date spans headed by legal terminology used as a modifier of the temporal expression. Indeed, it did not recognize the five date spans, which started with "on or about", or "at or about" which are legal terms of art.[18]

The FLAIR Rule-Modified (Silver) approach did not tag six of the eight phrases in the large test set that express periodic intervals (sets) such as *a nightly basis*.

5.3. Date Annotation with HeidelTime, SUTime & Claude Sonnet

The **date normalization** results by HeidelTime, SUTime, and Claude Sonnet 4.5 are summarized in Table 5. As shown in Tab. 6, HeidelTime and SUTime achieve moderate performance on the TempEval-3 and Tweets-M benchmarks (Ding et al., 2021). Our results on legal transcripts show further degradation, consistent with prior findings that rule-based normalization tools lack robustness in out-of-domain settings (Wang et al., 2016).

Navas-Loro et al. (2019) recognized that time complexities presented in the legal domain cannot fit within the TimeML annotation system because the document creation date may not be the only anchor date. Indeed, the anchor date can often be the date of the inciting incident that has led to the litigation rather than the document-creation date. Figure 4 provides a sentence in which the speaker references the day after the victim's date of death, June 12, 1994. In addition, when a lawyer is questioning a witness about a document serving as evidence in the case, such as an earlier letter or email, the creation date of the documentary evidence may be the anchor date. The below transcript excerpt provides an example of when the anchor date for the relative dates (shown in bold) is the creation date for the document being read into evidence.

> **And let's read this.** "My personal belief is that Enron stock is an incredible bargain at current prices, and we will look **back a couple of years from now** and see the great opportunity that we currently have." And that's the latter part of the quote where it mentions that you had been buying stock in **the last couple of months**.[19]

[18] "On or about" means "[a] phrase used in reciting the date of an occurrence or conveyance, to escape the necessity of being bound by the statement of an exact date." (Bla)

[19] https://www.famous-trials.com/enron/1803-laytestimony

	precision	recall	F1-score	#train	#val	#test
FLAIR and LL models						
FLAIR rule-modified (Silver data)	**.83**/.63	**.84**/.69	**.83**/.66	N/A	N/A	1,600/136
FLAIR OOTB	.77/.65	.78/.73	.78/.69	N/A	N/A	1,600/136
Claude Sonnet 4.5 (LLM)	.59/.64	.67/.76	.63/.70	N/A	N/A	1,600/136
finetuned models						
BERT cased (Silver data)	.74/**.68**	.83/**.78**	.78/**.72**	23,978	2,966	1,600/136
Legal BERT uncased (Silver data)	.75/.67	.78/.74	.76/.70	23,978	2,966	1,600/136

Table 4: Flat NER Task: Neural and generative model results (strict approach) of the date span identification task (without normalization); second number denotes mixed dataset from transcripts unrelated to large dataset and not used in the training of the finetuned models; Silver trained: denotes the large dataset training data distantly labeled by the FLAIR-modified rules

Back in July of 1994 **Date** , when you testified at the motion at the preliminary hearing in this mater, can you tell us if your memory was more fresh as to the events of June the 13th **Date** back in July of '94 **Date** than they are now?

Figure 4: Excerpt July 10, 1995 transcript; 2nd date span's anchor date is the victim's date of death on June 12, 1994

	doc			sent		
	p	r	F1	p	r	F1
large dataset						
HeidelTime	.59	.53	.56	.57	.52	.54
SUTime	**.63**	.62	.62	**.68**	**.64**	**.65**
Claude Sonnet 4.5	.61	**.65**	**.63**	.46	.48	.47
small dataset						
HeidelTime	.37	.24	.29	.42	.36	.39
SUTime	.41	.31	.35	.48	.49	.48
Claude Sonnet 4.5	**.65**	**.54**	**.59**	**.56**	**.46**	**.51**

Table 5: **Date normalization (strict):** based on 155 Gold large dataset samples and 136 Gold small dataset samples at the doc(ument) and sent(ence) level

	TempEval-3			Tweet-M		
	p	r	F1	p	r	F1
HeidelTime	.80	.76	.78	.88	.71	.79
SUTime	.68	.70	.69	.85	.88	.87

Table 6: End-to-end temporal expression normalization results for HeidelTime and SUTime on TempEval-3 and Tweets-M (Ding et al., 2021).

To address this issue, a temporal tagger must accommodate various anchor dates with a method of discerning among them, so that the correct anchor event can be identified and utilized in normalization. In contrast, the Claude Sonnet 4.5's results show that it could with prompting consider multiple anchor dates along with coreference issues when a speaker only partially referenced a previously provided date.

6. Conclusion

This research presents a new dataset for evaluating NER systems for spoken American English transcribed by court stenographers. While NER is an important natural language processing research area, there is sparse research on applying NER identification to text derived from speech. Researchers must provide more legal transcripts and English discourse datasets to refine NER abilities for verbal discourse. In addition, developing NER capabilities to identify Bates legal page numbering systems and evidentiary reference numbers would improve NER date-identifying models and provide additional capabilities for automating the review of legal transcripts. Moreover, the large dataset's trial transcript was from 1995 and thus preceded the general adoption of email and the age of texting. Future research should include additional legal transcripts from the last decade and models that can recognize whether relative dates are anchored to the transcript date or another date. New date annotation formats must be developed to accommodate multiple anchor dates. To develop timelines from the oral testimony contained in legal transcripts, which exceed current rule-based methods, one possible method is to build a model to derive normalized dates from the NER spans identified by our techniques. Lastly, other areas that must be explored for placing events contained in legal transcripts on a timeline are event coreference resolution and extracting temporal information by utilizing information in the discourse beyond a sentence (Adak et al., 2022), including model consideration of legal pleadings and other transcripts

from the legal matter for date and event coreference resolution.

7. Limitations

Our work is applied only to one large and several smaller legal transcripts. This is because we chose to focus primarily on identifying the full panoply of challenges introduced by legal transcripts. This approach allowed us to design annotation and automation strategies to begin to address the challenges posed by legal verbal discourse, which includes significant technical jargon and atypical speaking conventions.

We have not yet addressed the ordering of transcript excerpts according to their chronological date. We first wanted to test existing approaches (tools, rule-based approaches, and small and large language models) to understand the research and technology landscape. We will pursue in future work the chronological ordering of text derived from English legal transcripts, which is a task not yet resolved although it is posed in daily legal work.

8. Ethical Considerations

The authors hereby confirm that informed consent was obtained from human annotators. We did not process or store any personal information of human participants.

9. Bibliographical References

Black's law dictionary, 2nd ed. Accessed Oct 2025 at: http://thelawdictionary.org.

Sayantan Adak, Altaf Ahmad, Aditya Basu, and Mukherjee Animesh. 2022. Placing (historical) facts on a timeline: A classification cum coref resolution approach. In *Machine Learning and Knowledge Discovery in Databases*. Springer Nature Switzerland.

David Ahn, Sisay Fissaha, and Maarten de Rijke. 2005. Extracting temporal information from open domain text: A comparative exploration. In *5th Dutch-Belgian Information Retrieval Workshop*. https://pure.uva.nl/ws/files/4036502/38046_dir2005_timex.pdf.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. *In Proceedings of the 27th International Conference on Computational Linguistics*.

Ting Wai Terence Au, Ingemar J. Cox, and Vasileios Lampos. 2022. E-ner – an annotated named entity recognition corpus of legal text.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Angel X. Chang and Christopher Manning. 2012. SUTime: A library for recognizing and normalizing time expressions. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3735–3740, Istanbul, Turkey. European Language Resources Association (ELRA).

Scott Crossley, Yu Tian, Perpetual Baffour, Alex Franklin, Youngmeen Kim, Wesley Morris, Meg Benner, Aigner Picou, and Ulrich Boser. 2023. The english language learner insight, proficiency and skills evaluation (ellipse) corpus. *International Journal of Learner Corpus Research*, 9(2):248–269.

Leon Derczynski, Héctor Llorens, and Estela Saquete. 2012. Massively increasing TIMEX3 resources: A transduction approach. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3754–3761, Istanbul, Turkey. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Wentao Ding, Jianhao Chen, Jinmao Li, and Yuzhong Qu. 2021. Automatic rule generation for time expression normalization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3135–3144.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, Ralph M Weischedel, et al. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*, volume 2, pages 837–840. Lisbon.

Sylvain Galliano, Edouard Geoffrois, Djamel Mostefa, Khalid Choukri, Jean-François Bonastre, and Guillaume Gravier. 2005. The ester phase ii evaluation campaign for the rich transcription of french broadcast news. In *Interspeech*, pages 1149–1152.

Akash Gautam, Lukas Lange, and Jannik Strötgen. 2024. Discourse-aware in-context learning for temporal expression normalization. In *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, Mexico City, Mexico. Association for Computational Linguistics.

Erdal Kuzey, Vinay Setty, Jannik Strötgen, and Gerhard Weikum. 2016. As time goes by: Comprehensive tagging of textual phrases with temporal scopes. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 915–925, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

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, Dubrovnik, Croatia. Association for Computational Linguistics.

Wei Lu and Dan Roth. 2015. Joint mention extraction and classification with mention hypergraphs. In *Proceedings of EMNLP 2015*, pages 857–867, Lisbon, Portugal.

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.

Archana Patil, Shashikant Ghumbre, and Vahida Attar. 2023. Named entity recognition over dialog dataset using pre-trained transformers. In *Data Management, Analytics and Innovation*, Singapore. Springer Nature Singapore.

James Pustejovsky, José Castaño, Robert Ingria, Roser Saurí, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. Timeml: Robust specification of event and temporal expressions in text. In *IWCS-5. Fifth International Workshop on Computational Semantics*.

Emma Richardson, Kate Haworth, and Felicity Deamer. 2022. For the record: Questioning transcription processes in legal contexts. *Applied Linguistics 43/4: 677-697*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition.

Suwon Shon, Ankita Pasad, Felix Wu, Pablo Brusco, Yoav Artzi, Karen Livescu, and Kyu J Han. 2022. Slue: New benchmark tasks for spoken language understanding evaluation on natural speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7927–7931. IEEE.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, Istanbul, Turkey. European Language Resources Association (ELRA).

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

Kun Sun and Rong Wang. 2024. Automatic essay multi-dimensional scoring with fine-tuning and multiple regression. *ArXiv*.

Hegler Tissot, Angus Roberts, Leon Derczynski, Genevieve Gorrell, and Marcus Didonet Del Fabro. 2015. Analysis of temporal expressions annotated in clinical notes. In *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-11)*, London, UK. Association for Computational Linguistics.

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.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic. Association for Computational Linguistics.

Wei Wang, Kory Kreimeyer, Emily Jane Woo, Robert Ball, Matthew Foster, Abhishek Pandey, John Scott, and Taxiarchis Botsis. 2016. A new algorithmic approach for the extraction of temporal associations from clinical narratives with an application to medical product safety

surveillance reports. *J. of Biomedical Informatics*, 62(C):78–89.

Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Nianwen Xue, Martha Palmer, Jena D Hwang, Claire Bonial, et al. 2012. Ontonotes release 5.0. Accessed: Oct. 2025.

Mingli Wu, Wenjie Li, Qin Lu, and Baoli Li. 2005. CTEMP: A Chinese temporal parser for extracting and normalizing temporal information. In *Second International Joint Conference on Natural Language Processing: Full Papers*.

Juntao Yu, Bernd Bohnet, and Massimo Poesio. 2020. Named entity recognition as dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6470–6476, Online. Association for Computational Linguistics.

10. Appendix

10.1. Temporal Keyword Lists

Table 7 lists the temporal keyword vocabulary used for rule-based sentence filtering. Sentences containing at least one token matching these lists are flagged as candidate temporal expressions for further processing. The lists are applied via case-insensitive string matching.

10.2. Date Span Expansion via Temporal Modifiers

The FLAIR NER tagger identifies date spans at the token level using BIOES tags (e.g., `B-DATE`, `I-DATE`, `E-DATE`, `S-DATE`). However, temporal modifier words that are critical for interpreting the semantics of a date expression—such as *before*, *after*, *since*, or *during*—are frequently not included within the tagged date span. For example, in a securities litigation deposition, the sentence *"As part of your work for that retainer, you reviewed all the trading records during the first 20 days Dakota Plains traded?"* was tagged by FLAIR with only *the first 20 days* as a date entity, omitting the modifier *during*, which is essential for understanding the temporal scope of the inquiry.

To address this, we implemented a post-processing step that expands recognized date spans by searching for temporal modifier words within a window of up to three tokens before the head or after the tail of each FLAIR-tagged date span. Specifically, if any token within this three-token window appears in a predefined list of temporal modifiers (Table 9), the date span is extended from the original boundary to the position of the modifier, absorbing all intervening tokens. The BIOES tags are adjusted accordingly (e.g., an `E-DATE` tag is changed to `I-DATE` and the new boundary token receives the `B-DATE` or `E-DATE` tag). Because the window may contain multiple modifier words, a single expansion can incorporate more than one modifier. For example, in the sentence *"I haven't said a word on camera in about probably two months,"* FLAIR tagged only *two months*; both *about* and *probably* fall within the three-token look-behind window and are absorbed together, yielding the expanded span *about probably two months*. The intervening tokens between the modifier and the original span are also absorbed, as in *"received prior to July 3rd, 1995,"* where the modifier *prior* and the linking word *to* are both incorporated into the date span.

Table 8 presents examples from the evaluation datasets illustrating cases where modifiers were successfully incorporated as well as cases where the expansion produced false positives.

As shown in Table 8, the expansion works well when the modifier directly qualifies the temporal expression (e.g., *"two years before," "during the first 20 days,"* and *"prior to July 3rd"*). However, the method produces false positives when the modifier is used in a non-temporal sense or is syntactically part of an adjacent clause. For example, in *"a 911 call on May 21st after Mr. Depp left,"* the word *after* introduces a subordinate clause rather than modifying the date. Similarly, *"questions about the April 2nd examination"* uses *about* in the sense of "regarding" rather than "approximately." The word *from* is particularly prone to false positives in legal language, where it frequently indicates source or origin (e.g., *"photographs from the 1989 beating"*) rather than a temporal boundary. A dependency-parse-based approach verifying whether the modifier has a direct relationship to a token within the date span could potentially reduce these false positives, but we leave this refinement to future work.

10.3. Claude Prompts

10.3.1. TE Extraction Prompt

The TE extraction prompt shown at Figures 5 and 6 performs token-level sequence labeling on pre-tokenized sentences, assigning each token a BIOES tag (`B-DATE`, `I-DATE`, `E-DATE`, `S-DATE`, or `O`) to demarcate temporal expression span boundaries. The prompt ingests a JSON-serialized list of tokens for a single sentence and returns a JSON list of tags in one-to-one correspondence, enabling direct comparison with FLAIR NER span predictions and human-annotated gold labels using standard seqeval metrics. The prompt is sent to the Claude API (Sonnet 4.5, temperature=0, max_tokens=1024) for each test sentence.

Category	Keywords
Months	January, February, March, April, May, June, July, August, September, October, November, December, Jan, Feb, Mar, Apr, Jun, Jul, Aug, Sep, Oct, Nov, Dec
Days of week	Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Mon, Tue, Wed, Thur, Fri
Seasons	Spring, Fall, Autumn, Winter, Summer
Numeric ordinals	1st, 2nd, 3rd, 4th, 5th, 6th, 7th, 8th, 9th, 10th, 11th, 12th, 13th, 14th, 15th, 16th, 17th, 18th, 19th, 20th, 21st, 22nd, 23rd, 24th, 25th, 26th, 27th, 28th, 29th, 30th, 31st
Word ordinals	first, second, third, fourth, fifth, sixth, seventh, eighth, ninth, tenth, eleventh, twelfth, thirteenth, fourteenth, fifteenth, sixteenth, seventeenth, eighteenth, nineteenth, twentieth, twenty-first, twenty-second, twenty-third, twenty-fourth, twenty-fifth, twenty-sixth, twenty-seventh, twenty-eighth, twenty-ninth, thirtieth, thirty-first
Cardinal numbers	0–31
Year words	century, decades, era, year, month, day, week, weekend
4-digit years	1800–2999
2-digit years	00–10
Decades	1900s, 1950s, 1960s, 1970s, 1980s, 1990s, 2000s, 2010s, 2020s, aughts, 20s, 30s, 40s, 50s, 60s, 70s, 80s, 90s, 00s, twenties, thirties, forties, fifties, sixties, seventies, eighties, nineties
Holidays	New Year's Day, President's Day, Valentine's, Easter, Memorial Day, Arbor Day, Labor Day, Presidents' Day, Halloween, Thanksgiving, Christmas, XMAS

Table 7: Temporal keyword lists used for candidate sentence identification.

Sentence (excerpt)	FLAIR Span	Modif.	Corr.
"..., **two years [before]** the op-ed, **two years [before]** that ..."	two years	before	+
"...show that **months [before]** the op-ed ..."	months	before	+
"...trading records **[during] the first 20 days** ..."	the first 20 days	during	+
"...for **the year [after]** for the December 2012 ..."	the year	after	+
"...received **[prior] to July 3rd, 1995**."	July 3rd, 1995	prior	+
"...**[about probably] two months** ..."	two months	about, probably	+
"...a 911 call on **May 21st [after]** Mr. Depp left."	May 21st	after	–
"...FOR **SEVERAL MONTHS [DURING]** THE LENGTH OF THIS TRIAL ..."	SEVERAL MONTHS	DURING	–
"...questions **[about] the April 2nd** examination ..."	April 2nd	about	–
"...photographs **[from] the 1989** beating ..."	1989	from	–

Table 8: Examples of date span expansion. Bracketed bold words indicate tokens added by the post-processing step. + indicates a correctly expanded span; – indicates a false positive where the modifier belongs to an adjacent clause or is used in a non-temporal sense.

10.3.2. Sentence-Level Date Annotation Prompt

The prompt in Figure 7 and the transcript date is sent to the Claude API (Sonnet 4.5, temperature=0, max_tokens=4096) for each test sentence requiring date annotation at the sentence level.

Category	Modifier Words
Precedence	before, prior, until, by
Subsequence	after, since, post
Approximation	about, approximate, approximately, roughly, probably, likely
Range	between, during, from
Boundary	beginning, end, early, late
Degree	least, most

Table 9: Temporal modifier words used to expand FLAIR NER date spans. Both capitalized and lowercase forms are matched.

10.3.3. Document-Level Date Annotation Two-Level Prompt

Prompt 1: Document-Level Anchor Date Extraction The prompt in Figure 8 is sent once per transcript day to identify temporal anchor dates that serve as reference points for resolving relative temporal expressions. The known transcript date is prepended as a verified anchor. The full transcript text is appended (sampled to 30,000 characters if longer). Model configuration: Claude Sonnet 4.5, temperature=0, max_tokens=4,096.

Prompt 2: Document-Level Date Annotation The prompt in Figure 9 is sent for each batch of sentences (batch size = 10). All sentences in the batch are provided as full context, but only sentences belonging to the test split are marked for annotation. The anchor dates extracted by Prompt 1 and the transcript date are included as reference points for resolving relative temporal expressions. Model configuration: Claude Sonnet 4.5, temperature=0, max_tokens=4,096.

TE Extraction Prompt: BIOES Sequence Labeling (Instructions)

You are an expert at Named Entity Recognition for temporal expressions.
Your task: Tag each token with BIOES labels for DATE entities ONLY.

BIOES Scheme:
- B-DATE: Beginning of a date span (first token)
- I-DATE: Inside a date span (middle tokens)
- E-DATE: End of a date span (last token)
- S-DATE: Single token date (standalone)
- O: Not a date entity

DATE entities include:
- Absolute dates: "January 6th", "December 2020", "June 12th"
- Relative dates: "yesterday", "last week", "next month"
- Days of week: "Monday", "Wednesday"
- Months: "January", "February"
- Date components: years, days, months when referring to dates

CRITICAL: Include words of approximation and uncertainty WITH the date:
"I think it was Sunday" -> ["O","B-DATE","I-DATE","I-DATE","E-DATE"]
"approximately one day" -> ["B-DATE","I-DATE","E-DATE"]
"about three weeks" -> ["B-DATE","I-DATE","E-DATE"]
"roughly January 6th" -> ["B-DATE","I-DATE","E-DATE"]
"maybe Monday" -> ["B-DATE","E-DATE"]

CRITICAL: Include ARTICLES when part of the date phrase:
"the third of July" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"a Saturday in May" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"the 5th of January" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"the week of June 1st" -> ["B-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]

CRITICAL: Tag SPECIAL NAMED DURATIONS of a day or longer:
"Fiscal Year 2020" -> ["B-DATE","I-DATE","E-DATE"]
"His initial year in the program" -> "initial year" = ["B-DATE","E-DATE"]
"the first semester" -> ["B-DATE","I-DATE","E-DATE"]
"second quarter of 2021" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"spring term" -> ["B-DATE","E-DATE"]
"Q4 2023" -> ["B-DATE","E-DATE"]

CRITICAL: Tag AGES as temporal durations:
"She will be four years old on Tuesday"
-> "four years old" = ["B-DATE","I-DATE","E-DATE"], "Tuesday" = ["S-DATE"]
"He is 25 years old" -> ["O","O","B-DATE","I-DATE","E-DATE"]
"when I was 5 years old" -> ["O","O","O","B-DATE","I-DATE","E-DATE"]

CRITICAL: Tag DURATIONS >= 24 hours, INCLUDING modifiers:
"many years" -> ["B-DATE","E-DATE"]
"several months later" -> ["B-DATE","I-DATE","E-DATE"]
"a few weeks ago" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"a couple of days" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"slightly more than 3 days" -> ["B-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]

CRITICAL: Tag SETS (repeated/periodic times):
"once a week" -> ["B-DATE","I-DATE","E-DATE"]
"every Monday" -> ["B-DATE","E-DATE"]
"twice a day" -> ["B-DATE","I-DATE","E-DATE"]
"three times a month" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]

CRITICAL: Tag CONTINUOUS DATE RANGES as ONE span:
"between May 1, 2025 and April 20, 2026"
-> ["B-DATE","I-DATE","I-DATE","I-DATE","I-DATE",
"I-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]
"from January to March" -> ["B-DATE","I-DATE","I-DATE","E-DATE"]
"May 1st through June 30th" -> ["B-DATE","I-DATE","I-DATE",
"I-DATE","I-DATE","E-DATE"]

CRITICAL: Tag ALTERNATIVE/UNCERTAIN DATES as ONE span:
"the 5th or 6th of May" -> ["B-DATE","I-DATE","I-DATE",
"I-DATE","I-DATE","E-DATE"]
"January or February" -> ["B-DATE","I-DATE","E-DATE"]

CRITICAL: NO O TAGS INSIDE DATE SPANS:
WRONG: ["B-DATE","I-DATE","O","E-DATE"] X
CORRECT: ["B-DATE","I-DATE","I-DATE","E-DATE"] **DO NOT tag durations less than 24 hours:**
"3 hours" -> ["O","O"] "45 minutes" -> ["O","O"]

Figure 5: TE extraction prompt – instructions and annotation guidelines (Part 1 of 2). Sent once per sentence. Claude Sonnet 4.5, temperature=0, max_tokens=1,024.

TE Extraction Prompt: BIOES Sequence Labeling (Rules & Examples)

Instructions (summary):

1. Include approximation/uncertainty words as PART of the date entity
2. Include duration modifiers (many, several, few, a couple) with durations >= 24h
3. Include articles ("the", "a", "an") when part of the date phrase
4. Tag special named durations >= 1 day (Fiscal Year, semester, quarter, etc.)
5. Tag ages as temporal durations (four years old, 25 years old)
6. Approximation words + multiple dates with "or" = ONE continuous span
7. Tag durations >= 24 hours (days, weeks, months, years)
8. Tag sets/frequencies (once a week, every day, twice a month)
9. Tag continuous date ranges (between...and, from...to) as ONE span
10. Tag alternative/uncertain dates (with "or") as ONE span
11. NEVER place an O tag inside a date span
12. Tag multi-token dates with B-DATE, I-DATE, E-DATE
13. Tag single-token dates with S-DATE
14. Tag everything else as O
15. Return ONLY a JSON list of tags, one per token
16. The list must have EXACTLY {len(tokens)} tags

Examples:

Input: ["On","January","6th",",","2021"]
Output: ["O","B-DATE","E-DATE","O","S-DATE"]

Input: ["I","think","it","was","Sunday"]
Output: ["O","B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["It","took","approximately","one","day"]
Output: ["O","O","B-DATE","I-DATE","E-DATE"]

Input: ["It","took","slightly","more","than","3","days"]
Output: ["O","O","B-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["I","go","once","a","week"]
Output: ["O","O","B-DATE","I-DATE","E-DATE"]

Input: ["It","was","many","years","before","I","thought","of","Mary","again","."]
Output: ["O","O","B-DATE","E-DATE","O","O","O","O","O","O","O"]

Input: ["several","months","later"]
Output: ["B-DATE","I-DATE","E-DATE"]

Input: ["a","few","weeks","ago"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["a","couple","of","days"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["the","third","of","July"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["a","Saturday","in","May"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["Fiscal","Year","2020"]
Output: ["B-DATE","I-DATE","E-DATE"]

Input: ["the","first","semester"]
Output: ["B-DATE","I-DATE","E-DATE"]

Input: ["She","will","be","four","years","old","on","Tuesday","."]
Output: ["O","O","O","B-DATE","I-DATE","E-DATE","O","S-DATE","O"]

Input: ["He","is","25","years","old"]
Output: ["O","O","B-DATE","I-DATE","E-DATE"]

Input: ["between","May","1",",","2025","and","April","20",",","2026"]
Output: ["B-DATE","I-DATE","I-DATE","I-DATE","I-DATE","I-DATE", "I-DATE","I-DATE","I-DATE", "E-DATE"]

Input: ["from","January","to","March"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["the","5th","or","6th","of","May"]
Output: ["B-DATE","I-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["January","or","February"]
Output: ["B-DATE","I-DATE","E-DATE"]

Input: ["maybe","January","5th","or","February","10th"]
Output: ["B-DATE","I-DATE","I-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["January","6",",","2021"]
Output: ["B-DATE","I-DATE","I-DATE","E-DATE"]

Input: ["It","took","3","hours"]
Output: ["O","O","O","O"]

Now tag these tokens:
{json.dumps(tokens)}

Return ONLY the JSON list of tags, nothing else.

Figure 6: TE extraction prompt – enumerated rules and worked examples (Part 2 of 2). The penultimate line is populated at runtime with the pre-tokenized input sentence.

Sentence-Level Date Annotation Prompt

```
Extract any temporal expressions mentioned in the following text,
including dates, durations, sets/recurring periods, and fiscal years.
The transcript date is: {transcript_date}.

Format rules:

Dates – Use ISO 8601 format (YYYY-MM-DD).
      – If the year is unknown, use XXXX.
      – If the month is unknown, use XX.
      – If the day is unknown, use XX.
      – For date ranges, use START/END (e.g., 2012-07-01/2012-07-31).
      – For partial dates with unknown anchor plus offset,
            use XXXX-XX-XX/+P___ or XXXX-XX-XX/-P___.

Durations – Use ISO 8601 duration format: PnYnMnDTnHnMnS
      – Examples: P2Y (2 years), P20D (20 days), P6M (6 months),
            P1Y6M (1 year 6 months)
      – If the quantity is unknown/vague ("several months", "years"),
            use X: PXM, PXY, PXD
      – If a duration can be anchored to a concrete range,
            provide BOTH the ISO duration AND the resolved range.

Sets / Recurring periods – Use ISO 8601 repeating interval
      notation or codes:
      – EVERY-P1Y (every year), EVERY-P1M (every month),
            EVERY-P1W-05 (every Friday)

Fiscal years – Use FYyyyy (e.g., FY2011).
      – If the fiscal year is unknown, use FYXXXX.
      – If resolvable, provide BOTH: FY2011 and range
            (e.g., 2010-10-01/2011-09-30).

Day-of-week without year – Use XXXX-XX-XX-DD
      (01=Mon...07=Sun). Example: "on a Friday" -> XXXX-XX-XX-05.

If no temporal expression found, return an empty dates array.

Return ONLY valid JSON with this structure:
{
      "dates": [{
            "text": "exact temporal phrase from the sentence",
            "value": "YYYY-MM-DD or ISO duration or set notation",
            "value_resolved": "concrete date range if anchorable,
                        otherwise null",
            "type": "DATE | PARTIAL_DATE | RELATIVE_DATE |
                  DATE_RANGE | DURATION | SET |
                  FISCAL_YEAR | DATETIME | TIME",
            "confidence": "high | medium | low",
            "reasoning": "brief explanation"
      }]
}

Confidence guidelines:
      – high: explicitly stated ("January 5, 2020", "20 days", "FY2011")
      – medium: partially inferred ("in March" + year from anchor)
      – low: vague or ambiguous ("that summer", "for years")

Critical rules:
      1. Do NOT resolve durations without a clear anchor date.
            If uncertain, set value_resolved to null; keep ISO duration in value.
      2. Capture ALL temporal expressions, including purely durational ones.
      3. "the first 20 days" -> type=DURATION, value=P20D.
      4. "years" (vague) -> type=DURATION, value=PXY.
      5. Use your LLM, not python libraries.[4pt] Return only JSON, no explanation.
```

Figure 7: Sentence-level date annotation prompt sent to Claude Sonnet 4.5 (temperature=0, max_tokens=4,096) with the transcript date.

Prompt 1: Document-Level Anchor Date Extraction

```
KNOWN TRANSCRIPT DATE: {transcript_date}
(This date is already verified - use it as the primary
anchor)

Analyze this transcript and find anchor dates
(dates that serve as reference points).

IMPORTANT: Besides the transcript date, look for
OTHER anchor dates such as:
- Exhibits with dates (Exhibit A dated February 10,
2023)
- Events mentioned with dates (the meeting on
March 5th)
- Documents with dates (email dated January 15)

Return ONLY valid JSON:
{
      "transcript_date": {
            "date": "YYYY-MM-DD",
            "source": "where found"
      },
      "exhibits": [
            {
            "exhibit_id": "A",
            "date": "YYYY-MM-DD",
            "description": "brief"
            }
      ],
      "events": [
            {
            "event": "brief desc",
            "date": "YYYY-MM-DD"
            }
      ]
}

Transcript: {transcript_text}
```

Figure 8: Anchor date extraction prompt (Prompt 1) sent once per transcript day. Claude Sonnet 4.5, temperature=0, max_tokens=4,096.

Prompt 2: Document-Level Date Expression Annotation

```
Extract dates from the sentences marked for annotation
below.

ANCHOR DATES (use these as reference points):
PRIMARY ANCHOR - Transcript Date: {transcript_date}
(This is the verified date of the transcript -
use it to resolve relative dates)
OTHER ANCHORS - Exhibits: {exhibits_json}
OTHER ANCHORS - Events: {events_json}

FULL CONTEXT (for reference):
{all_sentences_in_batch}

ANNOTATE THESE SENTENCES ONLY:
{test_sentences_only}

Return ONLY valid JSON with this structure:
{
      "sentences": [
      {
            "index": 0,
            "dates": [
            {
                  "text": "February 10, 2023",
                  "type": "DATE",
                  "value": "2023-02-10",
                  "confidence": "high",
                  "reasoning": "explicit date
                  mentioned"
            }
            ]
            }
      ]
}

IMPORTANT:
- Use the transcript date and other anchors to resolve
relative dates (e.g., "the next day", "two days later")
- Only include indices for sentences marked ANNOTATE
above
- If no date, use empty dates array

Return only JSON, no explanation.
```

Figure 9: Document-level date annotation prompt (Prompt 2) sent per batch of 10 sentences. Claude Sonnet 4.5, temperature=0, max_tokens=4,096. Anchor dates from Prompt 1 are injected at runtime.