

Review

# Event-Centric Temporal Knowledge Graph Construction: A Survey

Timotej Knez \*  and Slavko Žitnik 

Faculty of Computer and Information Science, University of Ljubljana, 1000 Ljubljana, Slovenia; slavko.zitnik@fri.uni-lj.si

\* Correspondence: timotej.knez@fri.uni-lj.si

**Abstract:** Textual documents serve as representations of discussions on a variety of subjects. These discussions can vary in length and may encompass a range of events or factual information. Present trends in constructing knowledge bases primarily emphasize fact-based common sense reasoning, often overlooking the temporal dimension of events. Given the widespread presence of time-related information, addressing this temporal aspect could potentially enhance the quality of common-sense reasoning within existing knowledge graphs. In this comprehensive survey, we aim to identify and evaluate the key tasks involved in constructing temporal knowledge graphs centered around events. These tasks can be categorized into three main components: (a) event extraction, (b) the extraction of temporal relationships and attributes, and (c) the creation of event-based knowledge graphs and timelines. Our systematic review focuses on the examination of available datasets and language technologies for addressing these tasks. An in-depth comparison of various approaches reveals that the most promising results are achieved by employing state-of-the-art models leveraging large pre-trained language models. Despite the existence of multiple datasets, a noticeable gap exists in the availability of annotated data that could facilitate the development of comprehensive end-to-end models. Drawing insights from our findings, we engage in a discussion and propose four future directions for research in this domain. These directions encompass (a) the integration of pre-existing knowledge, (b) the development of end-to-end systems for constructing event-centric knowledge graphs, (c) the enhancement of knowledge graphs with event-centric information, and (d) the prediction of absolute temporal attributes.

**Citation:** Knez, T.; Žitnik, S.

Event-Centric Temporal Knowledge Graph Construction: A Survey.

*Mathematics* **2023**, *11*, 4852. <https://doi.org/10.3390/math11234852>

Academic Editor: Jie Wen

Received: 27 October 2023

Revised: 27 November 2023

Accepted: 30 November 2023

Published: 2 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** event extraction; temporal information; knowledge graphs; survey**MSC:** 68T35

## 1. Introduction

Understanding the discourse described in documents is a critical task for many AI systems that need to comprehend text. Understanding the discourse boils down to grasping the sequence of events that occur in a document and the people or things involved in those events. One particularly important part of events is their timing, as it helps us arrange the events in order of when they happened. The most common way to represent this information is by using knowledge graphs. Most research on creating knowledge graphs focuses on information about entities. This means that the knowledge graphs store facts about entities and how they are connected. Some well-known knowledge graphs that contain this kind of information are DBpedia [1], Wikidata [2], and YAGO [3]. These knowledge graphs often also include some information about events, especially historical events. For instance, Wang et al. added information about the timing of historical events to the YAGO knowledge base [4]. Such knowledge graphs are predominantly centered around entities and their attributes. However, in many real-world use cases, like clinical dead end prediction or automatic timeline reconstruction, it can be more important to

understand the discourse in text. For understanding such discourse, it is better to use a knowledge graph centered around events. In this type of graph, we represent events as nodes and the connections between events as edges. These connections mostly show the order in which events happened or how one event caused another. An interesting knowledge graph that contains common-sense information about events and entities is ATOMIC 2020 [5]. This graph contains common-sense information about how events are related. These relations cover more detailed information than just showing which event caused another event. The authors of the ATOMIC knowledge base also created a model that can predict these common-sense connections between events and entities that are not in the knowledge base. Besides events and their connections, knowledge graphs also often include other things that describe the time-related aspects of events.

Building knowledge graphs focused on events typically involves a series of smaller tasks. A general overview of how a graph is constructed is shown in Figure 1. There are three main steps: (1) figuring out what the events are and gathering information about them, (2) finding out how events are related in terms of time and cause and effect, and (3) creating a knowledge graph from all this information. Some methods combine the steps of figuring out events and their relationships into one step to make a more straightforward process. This is most common in the medical field because events there usually do not need additional details. Some models also use extra knowledge when figuring out the timing of events. This extra information can make the models work better in some cases. Instead of using a knowledge graph, some previous studies represent events in timelines, which are simpler but can still give a good idea of how the events occurred in a document.

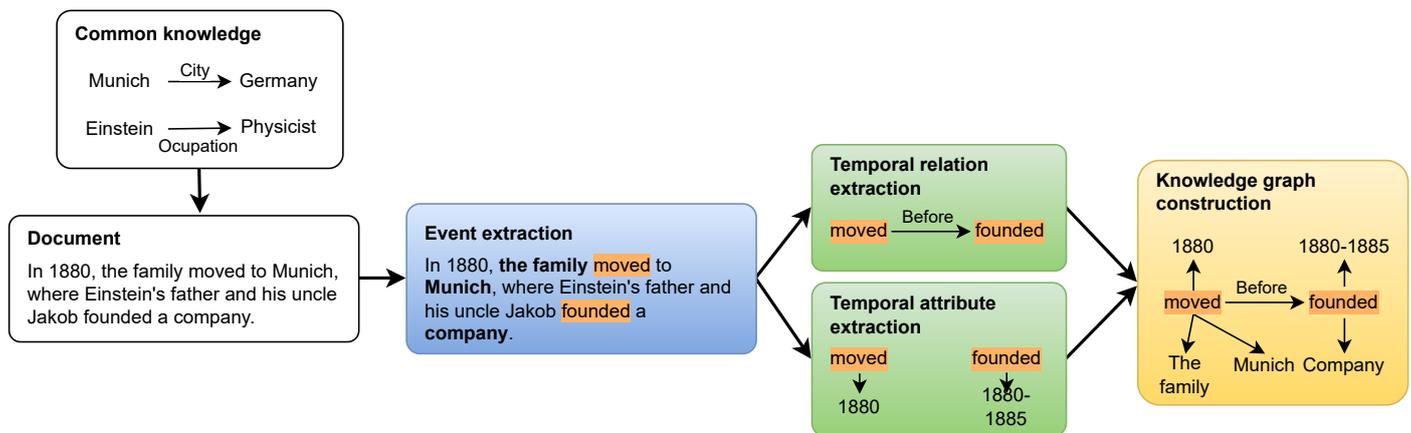


Figure 1. General pipeline for event-centric temporal knowledge graph construction.

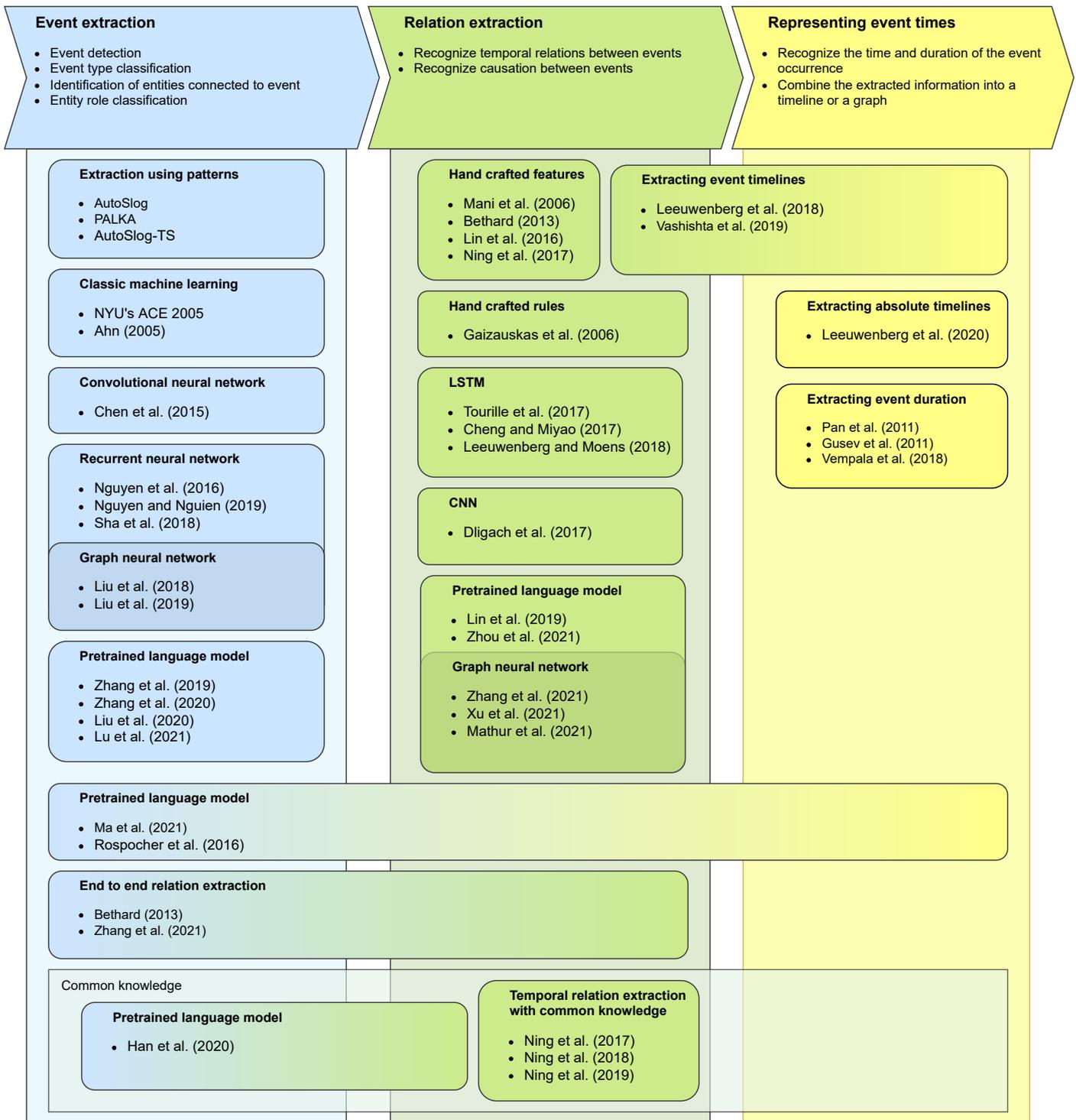
In this survey, we provide an overview of the primary methods employed to address the tasks essential for creating knowledge graphs centered around events using unstructured documents. Figure 2 illustrates the systems discussed in our paper, positioning them based on their key characteristics. We assess these systems according to whether they engage in event extraction, temporal relation extraction, event duration identification, or event timeline generation. Notably, only two of the systems we present encompass the complete process, spanning from event extraction to knowledge base construction. We cluster the systems into smaller groups, based on the machine learning techniques employed (see Figure 2).

We introduce systems tailored to extract events and their associated attributes, temporal relationships, and additional temporal characteristics, ultimately culminating in the development of event-centric knowledge graphs or timelines. A comprehensive comparative analysis of all the systems featured herein is presented in Table 1. Our overarching objective is to guide future research endeavors toward the automated construction of temporal event-centric knowledge graphs. We achieve this by delineating existing work in

this domain and discussing the remaining challenges and opportunities that need to be addressed to enhance the efficacy of such systems.

**Table 1.** Comparison of features from all of the described projects. EE: event extraction, EA: event attribute extraction, TR: temporal relation extraction, ED: event duration recognition, TC: time-line creation, add: additional attributes, SY: syntactic dependencies, CL: cross-lingual model, CK: common knowledge.

Paper	EE	EA	TR	ED	TC	Add.	Domain
Riloff (1993) [6]	✓	✓	✗	✗	✗		News
Riloff (1995) [7]	✓	✓	✗	✗	✗		News
Kim and Moldovan (1995) [8]	✓	✓	✗	✗	✗		News
Grishman et al. (2005) [9]	✓	✓	✗	✗	✗		News
Ahn (2006) [10]	✓	✓	✗	✗	✗		News
Yang and Mitchell (2016) [11]	✓	✓	✗	✗	✗		News
Chen et al. (2015) [12]	✓	✓	✗	✗	✗		News
Nguyen et al. (2016) [13]	✓	✓	✗	✗	✗		News
Sha et al. (2018) [14]	✓	✓	✗	✗	✗	SY	News
Liu et al. (2018) [15]	✓	✓	✗	✗	✗	SY	News
Liu et al. (2019) [16]	✓	✓	✗	✗	✗	SY, CL	News
Zhang et al. (2019) [17]	✓	✓	✗	✗	✗		News
Zhang et al. (2020) [18]	✗	✓	✗	✗	✗		News
Ji (2009) [19]	✓	✓	✗	✗	✗	CL	News
Zhu et al. (2014) [20]	✓	✓	✗	✗	✗	CL	News
Chen et al. (2009) [21]	✓	✓	✗	✗	✗	CL	News
Liu et al. (2020) [22]	✓	✓	✗	✗	✗		News
Lu et al. (2021) [23]	✓	✓	✗	✗	✗		News
Gaizauskas et al. (2006) [24]	✗	✗	✓	✗	✗		News
Mani et al. (2006) [25]	✗	✗	✓	✗	✗		News
Bethard (2013) [26]	✓	✗	✓	✗	✗		News
Lin et al. (2016) [27]	✗	✗	✓	✗	✗		Medical
Ning et al. (2017) [28]	✗	✗	✓	✗	✗	CK	News
Tourille et al. (2017) [29]	✗	✗	✓	✗	✗		Medical
Dligach et al. (2017) [30]	✗	✗	✓	✗	✗		Medical
Lin et al. (2019) [31]	✗	✗	✓	✗	✗		Medical
Cheng and Miyao (2017) [32]	✗	✗	✓	✗	✗	SY	News
Leeuwenberg and Moens (2018) [33]	✗	✗	✓	✓	✓		News
Zhou et al. (2021) [34]	✗	✗	✓	✗	✗	CL	Medical
Zhang et al. (2021) [35]	✓	✗	✓	✗	✗	SY	News
Xu et al. (2021) [36]	✗	✗	✓	✗	✗	SY	News
Mathur et al. (2021) [37]	✗	✗	✓	✗	✗	SY	News
Ning et al. (2018) [38]	✗	✗	✓	✗	✗	CK	News
Ning et al. (2019) [39]	✗	✗	✓	✗	✗	CK	News
Han et al. (2020) [40]	✓	✗	✓	✗	✗	CK	News, Medical
Leeuwenberg and Moens (2020) [41]	✗	✗	✗	✓	✓		Medical
Vashishtha et al. (2019) [42]	✗	✗	✓	✓	✓		News
Pan et al. (2011) [43]	✗	✗	✗	✓	✗		News
Gusev et al. (2011) [44]	✗	✗	✗	✓	✗		News
Vempala et al. (2018) [45]	✗	✗	✗	✓	✗		News
Rospocher et al. (2016) [46]	✓	✓	✓	✗	✓		News
Ma et al. (2021) [47]	✓	✓	✓	✓	✓		News
Jindal and Roth (2013) [48]	✓	✗	✗	✗	✗		Medical



**Figure 2.** The systems performing main tasks required for event-centric knowledge graph construction [6–10,12–18,22–47,49].

The main contributions of this work are summarized as follows:

- (a) **Literature review for all tasks related to temporal event knowledge graph construction.** We identify the tasks required for the automatic construction of event-centric temporal knowledge graphs and provide a review of research dealing with each of the tasks. We compare different approaches and identify the most successful model designs.

- (b) **Identification of emerging advancements.** We identify the directions in which systems for each of the tasks are likely to evolve in the future. We also provide our opinion on which approaches seem to be the most prospective for future systems.
- (c) **Identification and proposal of open research problems.** We summarize each of the areas and highlight promising future research directions. We also provide the main criteria that future work should achieve.

We identified several surveys that address general relation extraction [50–55]. In contrast, there are fewer surveys dedicated to the topics of event extraction [56] and temporal relation extraction [57,58]. Additionally, we found surveys that explore knowledge graphs [59,60].

To the best of our knowledge, our survey paper represents the first comprehensive examination of the entire process involved in constructing temporal event-centric knowledge graphs. While previous surveys have delved into individual aspects of this process, none have provided a comprehensive overview of all the challenges that must be addressed to facilitate the automated creation of such knowledge graphs.

The remainder of this paper is structured as follows: In Section 3, we elucidate the methodologies employed in conducting our survey. Section 4 details the existing standards for representing events, their relationships, and associated properties. We also present datasets containing temporal information about events in documents. Section 5 offers an overview and comparison of existing models for event extraction. In Section 6, we present models designed for the extraction of temporal relations. Subsequently, in Section 7, we introduce existing systems capable of constructing event timelines and knowledge graphs. Finally, Sections 8 and 9 delve into prospects for future research in this domain and provide a concluding perspective on the paper’s findings.

#### *Formal Definition of an Event-Centric Knowledge Graph*

An event-centric temporal knowledge graph is a structured representation of knowledge that models and organizes information about events, their attributes, and their temporal relationships within a domain. It is composed of a directed graph consisting of nodes and edges, where nodes represent events and their associated attributes, and edges represent temporal relationships between events. The knowledge graph is designed to capture the temporal nature of events, allowing for the representation of chronological sequences, durations, and temporal overlaps [61].

We define an event-centric temporal knowledge graph with the following components:

1. **Nodes:** The knowledge graph is comprised of a set of nodes  $V = \{v_1, v_2, \dots, v_n\}$  representing individual events. Each node  $v_i$  corresponds to a specific event.
2. **Attributes:** Each node  $v_i$  contains a set of attributes associated with the event. We denote the set of attributes as  $A(v_i) = \{a_1, a_2, \dots, a_m\}$ .
3. **Edges:** The knowledge graph contains a set of directed edges between nodes  $v_i$  and  $v_j$ , where each edge represents a temporal relation. We denote a set of edges as  $E = \{e_1, e_2, \dots, e_k\}$ , where each edge  $e_i = (v_1, v_2, r)$  is a triplet of two nodes and a relation  $r$  from the set of valid relations  $R$ , which differs between different domains and use cases.

Event-centric temporal knowledge graphs have a number of properties that need to hold for a valid graph:

1. **Temporal consistency:** A temporal knowledge graph ensures consistency between temporal relations so that events with temporal relations form valid timelines.
2. **Granularity:** An event-centric knowledge graph needs to accommodate different levels of event granularities. An event might be a specific instance that occurred at a specific point in time or a general concept that can occur in multiple situations.

## 2. Motivation

In the rapidly evolving landscape of information, the ability to extract temporal sequences from unstructured documents holds immense potential for unlocking valuable insights. This survey explores innovative approaches that empower the recognition of temporal patterns, laying the groundwork for constructing event timelines and unraveling the dynamics of how events unfold. Extracting temporal information allows us to create a structured representation of the chronological order in which events occur. This chronological perspective is invaluable for understanding the evolution of various phenomena and gaining insights into the underlying mechanisms.

The extraction of temporal sequences facilitates the identification of patterns and trends within corpora. By discerning the temporal relationships between events, we can unveil hidden correlations, enabling us to make informed predictions and strategic decisions. This capability is useful in diverse domains, such as customer behavior analysis, where the identification of patterns can be harnessed for applications like recommender systems [62].

Temporal knowledge graphs are also a powerful tool with interdisciplinary applications. Beyond the realms of computer science, they find utility in social sciences, history, disaster management, and beyond. These knowledge graphs serve as dynamic repositories of temporal information, fostering a deeper understanding of complex relationships and interactions across diverse domains. An example of such use is trend analysis as demonstrated by Bonifazi et al. [63]. Their exploration of time patterns in TikTok trends not only sheds light on emerging trends but also helps discern potentially harmful trends. This analytical approach is instrumental in staying ahead of the curve in dynamic environments. Recent developments in sentiment scope analysis, as highlighted by Bonifazi et al. [64], showcase another facet of the importance of temporal sequence extraction. Understanding the temporal scope of a user's sentiment on a topic in a social platform enables a nuanced understanding of evolving opinions and attitudes over time.

The extraction of temporal sequences from documents transcends disciplinary boundaries, offering a versatile toolkit for understanding, predicting, and influencing various phenomena. From constructing event timelines to uncovering hidden patterns and trends, the applications are diverse and impactful, making this field a focal point of research and development across multiple domains.

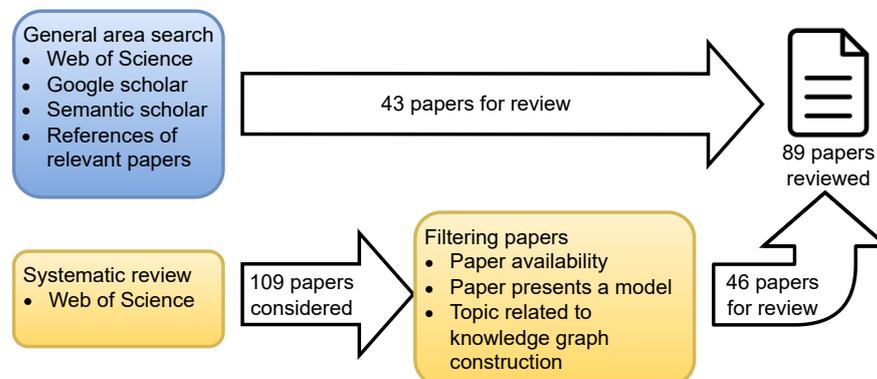
## 3. Methodology

The objective of this review is to present the primary methodologies for automating the construction of event-centric temporal knowledge graphs. We identified event extraction, temporal relation, and attribute extraction, as well as knowledge graph or timeline construction as the three main steps required for constructing an event-centered knowledge graph. These tasks were determined through an analysis of existing models that are capable of automatically generating event-centric temporal graphs or timelines.

Our selection of papers for inclusion in this survey adhered to a systematic methodology as illustrated in Figure 3. Initially, we conducted queries on scientific search engines Google Scholar, Semantic Scholar, and Web of Science, using general queries pertaining to the creation of event-centric knowledge graphs. Subsequently, we scrutinized the top search results to gain a foundational understanding of the field. Additionally, we considered papers that were referenced in the papers we identified as pertinent to our survey.

Upon acquiring a comprehensive understanding of the tasks associated with extracting temporal information about events and the primary strategies for addressing these tasks, we systematically reviewed papers related to the identified tasks. To accomplish this, we utilized the Web of Science database, executing a distinct query for each task. For event extraction, we searched for papers featuring the terms "event" and "extraction" in their titles. Given that many returned papers were not directly relevant, we refined the query by stipulating that the paper also includes the phrase "natural language processing". This modification ensured that the majority of retrieved papers were germane to our survey.

In the case of temporal relation extraction, we sought papers with titles containing the words “temporal”, “relation”, and “extraction”. For the extraction of event-centric knowledge graphs, we identified papers with titles comprising the terms “event”, “knowledge graph” or “timeline”, and “extraction” or “construction”.



**Figure 3.** Flow diagram of our systematic review process.

Our selection criteria for including papers in this survey were based on specific considerations. Firstly, a paper had to present a functional model or pipeline. Secondly, the task addressed in the paper needed to be directly related to the construction of event-centric temporal knowledge graphs. Furthermore, we excluded papers that had already been included in our preliminary review. During the systematic phase of the review, we assessed a total of 109 papers, ultimately incorporating 46 papers into our survey.

In addition to discussing information extraction models, we also introduce datasets containing temporal information about events. These datasets can prove invaluable for the development of future models. We identified these relevant datasets by analyzing the results provided by Google Scholar and examining the datasets employed in papers outlining pertinent models.

#### 4. Schemas and Datasets

For the tasks of event, temporal relation, and temporal attribute extraction, we need datasets that we can use to train machine learning models. To create the datasets, we need to define annotation schemas for annotating the information we would like to capture. In this section, we describe the schemas used to present events and their information.

##### 4.1. Schemas for Capturing Events

In the realm of event extraction, two prominent event schemas have emerged to structure and annotate textual information: the TimeML model [65] and the ACE (Automatic Content Extraction) model [66]. Each schema serves a distinct purpose in capturing information about events, with its own characteristics:

##### TimeML Model:

**Aim:** The TimeML model seeks to identify and capture all mentions of events within the text and establish relations between these events.

**Coverage:** It aims for comprehensive event recognition, encompassing a broad spectrum of event types.

**Annotations:** TimeML employs XML tags to annotate text, marking event mentions and their attributes, temporal expressions, their meanings, and relationships between them.

**Additional Information:** Documents in TimeML format often incorporate the recording of document creation times as a special time expression.

**ACE (Automatic Content Extraction) Model:**

**Aim:** The primary objective of the ACE event model is to provide more detailed information about each event, focusing on a predefined set of event types.

**Coverage:** ACE defines a specific set of 34 event types, and the model is designed to capture events that fall within these predetermined categories.

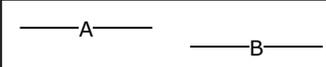
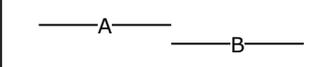
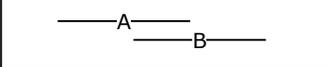
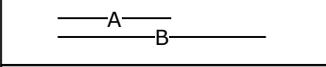
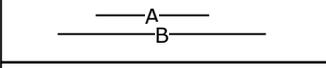
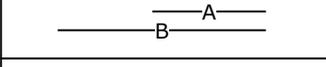
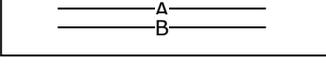
**Annotations:** ACE specifies a structured format for annotating text with details about events, including their attributes.

**Attributes:** Each event type in the ACE model comes with a predefined set of attributes, which vary depending on the specific event type.

These schemas play a vital role in the annotation of text documents to extract information about events and time expressions. The TimeML specification, in particular, has served as a foundation for the annotation format in numerous time-centric corpora, such as TimeBank [67], AQUAINT [68], i2b2 [69], and WikiWars [70]. By adopting these standardized schemas and formats, researchers can effectively annotate text data to facilitate event-centric knowledge graph construction and temporal information extraction.

*4.2. Schemas for Capturing Temporal Relations*

Temporal relations are a fundamental aspect of representing the temporal dynamics of events in a document. They provide crucial information about the sequence in which events occurred. Among the various temporal relations, some of those most commonly of interest include before, after, and during. In 1989, Allen and Hayes [71] defined a comprehensive set of 13 possible relations between two time intervals as illustrated in Figure 4. However, in practice, many temporal relation extraction datasets focus on annotating a subset of Allen’s relations that are most pertinent to the specific task or domain under consideration. These annotated datasets play a pivotal role in training and evaluating machine learning models for temporal relation extraction and event-centric knowledge graph construction.

	A precedes B	B is preceded by A
	A meets B	B is met by A
	A overlaps with B	B is overlapped by A
	A starts B	B is started by A
	A during B	B contains A
	A finishes B	B is finished by A
	A is equal to B	B is equal to A

**Figure 4.** Temporal relations between events as defined by Allen and Hayes [71].

The TimeML schema [65,72,73] serves as the most prevalent schema for representing temporal relations between events. It offers a structured framework for encoding basic temporal relations such as “Before”, “After”, and “During”.

In addition to these fundamental temporal relations, there has been research dedicated to extracting more precise relations that encompass estimates of the time intervals, event durations, and related temporal attributes [33,42]. These precise relations are particularly valuable for constructing event timelines and demand a distinct annotation schema. Unlike

the standardized TimeML schema, projects focusing on these precise relations often define their own schemas tailored to their specific requirements and objectives. As a result, there is no unified schema for representing these intricate temporal relationships.

#### 4.3. Capturing Temporal Properties of Events

In the context of describing the time of an event, it is imperative to define the specific time properties of interest. For that purpose, we analyzed the properties that are defined in the OWL time ontology (<https://www.w3.org/TR/owl-time/> (accessed on 12 October 2023)), as it is commonly used to record the time in knowledge graphs. The OWL time ontology encapsulates several key time components, including (1) time of the event (date, year, month, day, day of the week, time zone, etc.), (2) duration (years, months, weeks, days, etc.), (3) interval (combination of a time and duration), and (4) temporal relation (before, after, meets, met by, overlaps overlapped by, starts, started by, during, contains, finishes, finished by, and equals). We found that in addition to the temporal properties defined in the time ontology, it would also be useful to track the repetition of an event. For example, an event of taking medication might repeat every 24 h for a set amount of time. Leeuwenberg and Moens [41] provided annotations for event times and durations. The prediction of event duration has also been explored in multiple other systems (presented in Section 7.3). Temporal attributes other than the time and duration of an event remain unexplored.

#### 4.4. General Datasets for Temporal Description and Relation Extraction

One of the earliest temporal-centric corpora introduced to the field is the TimeBank 1.1 corpus [74]. This corpus comprises 186 documents extracted from the domain of news articles. Annotations in this corpus adhere to the TimeML standard, version 1.1 [65], and were established as a proof of concept for the TimeML specification [67]. However, the TimeBank 1.1 corpus harbors certain inconsistencies and errors, leading to its replacement by version 1.2, known as the TimeBank 1.2 corpus [75]. The TimeBank 1.2 corpus adheres to the updated TimeML specification [72] and encompasses 183 news articles, encompassing approximately 61,000 tokens. This dataset encompasses texts featuring annotated time expressions, events, and temporal relations between the events. In addition to conforming to the TimeML standard, the corpus includes supplementary information, such as sentence separation. In parallel to the TimeBank corpus, the TimeML specification was also leveraged to construct the AQUAINT corpus [68]. The AQUAINT corpus consists of 73 English news reports and bears structural similarities to the TimeBank corpus.

Notably, these datasets, which facilitate the extraction of temporal information from textual documents, have been featured in the SemEval shared tasks. The inaugural shared task TempEval took place during the 2007 SemEval challenge [76] and utilized a dataset derived from the TimeBank 1.2 corpus. However, it included only a subset of the original dataset's relations and annotations. A follow-up task, TempEval-2, was conducted in 2010 [77], employing a dataset based on the TimeBank 1.2 corpus but featuring distinct annotated relations. In 2013, SemEval introduced TempEval-3 [78], aimed at enabling models that leverage large neural networks. For this purpose, the organizers provided an extensive dataset conforming to the TimeML annotation specification. They augmented the TimeBank 1.2 dataset by incorporating missing relations and annotations. This dataset served as both the gold and evaluation components. Additionally, a substantial amount of text from the English Gigaword corpus [79] was automatically annotated to generate a sizable "silver" dataset. The silver dataset, while potentially containing errors, primarily serves as training data for large neural models. Notably, this corpus introduces over 600 thousand new silver tokens, 20 thousand new gold tokens, and 20 thousand new evaluation tokens.

One limitation of the TimeBank corpus is the relatively sparse occurrence of temporal relations among event pairs. To address this, Cassidy et al. [80] developed the TimeBank-Dense dataset. This dataset entails manual re-annotation of temporal relations between a substantially larger number of event pairs, effectively mitigating the sparsity issue. While

the dataset comprises only 36 documents, the overall number of annotated relations is four times greater than those in the TimeBank corpus. Nevertheless, the TimeBank-Dense dataset still grapples with low inter-annotator agreement as identified by Ning et al. [38]. In response, Ning et al. [38] introduced the MATRES dataset. In MATRES, temporal relations between events are based solely on the starting point of the event. Notably, MATRES employs multiple axes to compare events to one another, enhancing the precision of temporal annotations.

In 2017, Zhong et al. [81] introduced the Tweets dataset, annotating time expressions in concise text documents sourced from the Twitter social network. This dataset encompasses 942 documents and 18 thousand tokens, covering a range of general topics rather than focusing on a specific domain. Notably, Twitter documents differ from other similar datasets in their brevity and inclusion of precise time and date of posting.

We present the comparison of the available datasets in Table 2. The most commonly used datasets for temporal information extraction from news articles are the dataset provided for the TempEval-3 challenge, TimeBank-Dense and the MATRES dataset. The TempEval-3 dataset is useful for its large number of documents, enabling the use of neural network models. Its main drawback is the sparsity and low quality of annotated relations. These problems are addressed in the TimeBank-Dense and MATRES datasets. While the number of documents in this dataset is significantly smaller, the annotated relations are a lot more complete. We believe that the MATRES dataset is better than TimeBank-Dense for most applications, due to its higher inter-annotator agreement.

**Table 2.** The datasets on general domains for extracting temporal expressions and temporal event relations.

Dataset	Year	Domain	Amount of Data	Documents Origin
TimeBank 1.1	2003	News	186 documents	Original
TimeBank 1.2	2006	News	183 documents	Original
AQUAINT	2002	News	73 documents	Original
TempEval	2007	News	183 documents	TimeBank 1.2
TempEval-2	2010	News	183 documents	TimeBank 1.2
			61 k TimeBank tokens	
			34 k AQUAINT tokens	TimeBank 1.2
TempEval-3	2013	News	666 k new silver tokens	AQUAINT
			20 k new gold tokens	Gigaword
			20 k new evaluation tokens	
TB-Dense	2014	News	36 documents	TimeBank 1.2
MATRES	2018	News	36 documents	TimeBank-Dense
Tweets	2017	Twitter	942 documents (18k tokens)	Original

#### 4.5. Domain Specific Datasets

The task of temporal information extraction finds applicability across diverse domains, prompting the development of domain-specific datasets aimed at capturing unique temporal information within those domains.

One such domain-specific dataset is WikiWars, introduced in 2010 [70]. This corpus comprises Wikipedia articles centered around famous wars. Within these documents, time expressions are meticulously annotated in the TIMEX2 format (a subset of the TimeML annotation schema). The WikiWars dataset encompasses 22 documents, containing nearly 120 thousand tokens and featuring 2671 annotated temporal expressions. Another domain-specific temporal dataset emerges from the i2b2 2012 challenge [69], catering to the medical domain. This dataset supplies 310 discharge summaries replete with annotated time expressions, events, and the temporal relations existing between these events. Notably, the temporal relations in this dataset exhibit greater granularity, with eight distinct relation types: “before”, “after”, “simultaneous”, “overlap”, “begun by”, “ended by”, “during”, and “before with overlap”.

Expanding further into the medical extraction of temporal information, the THYME dataset emerged for the 2015 Medical TempEval task [82]. This dataset, drawn from

clinical notes on patients with colon cancer from the Mayo Clinic, comprises approximately 600 documents, each painstakingly annotated. Annotations encompass marked events, time expressions, and temporal relations between events. The same task was subsequently revisited in SemEval 2016 [83] using the THYME dataset. In this iteration, the training and testing segments of the SemEval 2015 dataset were employed as the training set, with new test data introduced.

SemEval 2017 saw the reappearance of the Clinical TempEval task [84], this time with a focus on evaluating the transfer learning capabilities of temporal information extraction systems. Models were trained on clinical notes related to colon cancer patients from the 2016 Clinical TempEval task and subsequently tested on clinical notes pertaining to patients with brain cancer. The dataset designed for this task encompasses 591 documents on colon cancer patients and 595 documents on brain cancer patients. Annotations within these documents mirror those employed in previous Clinical TempEval tasks, ensuring consistency across evaluations and facilitating the exploration of transfer learning in the context of temporal information extraction.

A comparative analysis of datasets specific to various domains is provided in Table 3. Predominantly, the medical domain emerges as the most frequently explored domain in the context of temporal relation extraction. Within this domain, two primary datasets stand out: the i2b2 2012 dataset and the THYME dataset. The THYME dataset exhibits notable advantages in terms of its size, rendering it particularly advantageous for training machine learning models with a substantial number of parameters. Consequently, it is considered a robust dataset for training models geared toward temporal relation extraction. Conversely, the i2b2 2012 dataset offers additional value by being annotated with supplementary temporal attributes by Leeuwenberg and Moens [41]. This enrichment extends its utility beyond temporal relation extraction, encompassing the extraction of absolute temporal attributes, such as duration and event times. Importantly, all the presented datasets adhere to the TimeML annotation schema, facilitating the possibility of combining multiple datasets to train a single model. This approach can be particularly advantageous for training larger neural network models, leveraging the diversity of data sources to enhance model performance.

**Table 3.** The datasets on specific domains for extracting temporal expressions and temporal event relations.

Dataset	Year	Domain	Amount of Data
WikiWars	2010	Wikipedia articles about famous wars	22 documents 120 k tokens
i2b2 2012	2012	Discharge summaries	310 documents 178 k tokens
THYME (TempEval 2015)	2015	Notes on colon cancer	440 documents
THYME (TempEval 2016)	2016	Notes on colon cancer	600 documents
THYME (TempEval 2017)	2017	Notes on colon and brain cancer	591 colon documents 595 brain documents

#### 4.6. Datasets for More Precise Temporal Relation Extraction

Several researchers have dedicated their efforts to establishing more fine-grained temporal relations between events, enabling the construction of comprehensive timelines within documents. These relations are not restricted to a limited set of categories (e.g., “before” and “after”) but instead are represented as continuous values that denote the precise chronological order of events. To facilitate the training of models for this purpose, multiple datasets have been developed. A comparison of datasets for precise temporal relations is provided in Table 4.

The Event StoryLine Corpus [85] is one such dataset, designed to facilitate the creation of storylines from news articles. A unique challenge addressed by this dataset is the

detection of event co-occurrence, allowing for the identification and prevention of event duplication when the same event is mentioned in the text.

The Fine-grained Temporal Relations dataset [42] endeavors to construct event timelines by going beyond the coarse temporal relations explored in previous datasets. This dataset not only recognizes coarse temporal relations but also identifies event durations and overlaps. This additional information facilitates the placement of each event on a relative timeline. Notably, the corpus employs a distinct annotation schema, departing from the TimeML standard. It comprises 91 thousand event pairs positioned on a relative timeline. The motivation for building event timelines also underlies the dataset developed by Leeuwenberg and Moens [41]. This dataset incorporates documents from the 2012 i2b2 challenge, specifically medical discharge summaries featuring labeled events. Within this dataset, absolute start times, end times, and event durations are annotated. Recognizing that precise absolute times and durations are often unexpressed in documents, the dataset provides upper and lower bounds for each time or duration.

Many existing datasets primarily encompass relations between events mentioned within the same or neighboring sentences. However, there is a need to discern temporal relations between events that appear in different sections of a document. To support systems capable of extracting such distant relations, the TDDiscourse dataset [86] was conceived. This dataset incorporates the same documents employed in the TimeBank-Dense dataset and comprises relations between events separated by more than one sentence in the original document. It comprises 6 thousand manually annotated relations and an additional 38 thousand relations automatically extracted. Another dataset catering to the extraction of relations between distant events is the Cross-document Event Corpus [87]. This corpus includes news documents sourced from the ACE2005 dataset, supplemented by additional news documents. In total, the dataset encompasses 125 documents and features 26 thousand event pairs.

**Table 4.** The datasets for more advanced tasks in temporal information extraction.

Dataset	Year	Domain	Amount of Data
Event StoryLine Corpus	2018	News articles	281 documents
Fine-grained temporal relations [42]	2019	English Web Treebank	250 k tokens
Leeuwenberg and Moens [41]	2020	Discharge summaries	310 summaries
TDDiscourse	2019	News articles	36 documents
Cross-document event corpus	2016	News articles	125 documents

The datasets presented in Table 4 fulfill two main goals. The first three datasets enable the extraction of event timelines instead of extracting individual temporal relations. The last two datasets, on the other hand, aim to enable relation and event extraction across an entire document or even multiple documents. When working with event timelines, we believe that the most interesting dataset is the one provided by Leeuwenberg and Moens since it contains absolute times of the events, while the other two datasets only contain relative positions of events on a timeline. The absolute times enable a number of additional applications that are not possible when only working with relative timelines.

#### 4.7. Temporal Extraction for Non-English Languages

Most of the available datasets for temporal extraction contain English texts. Some other languages, however, might contain different properties that make models trained on English perform poorly. To enable the training of models designed to work in a non-English language or to work in multiple different languages, some additional datasets have been released.

In 2005, the ACE 2005 Multilingual Training Corpus was released [66]. The corpus focuses on event extraction tasks but also contains temporal attributes. The ACE corpus contains documents in three different languages. The included languages are English, Arabic, and Chinese. The documents came from multiple different sources and include newswire,

broadcast news, conversation, weblog, discussion forums, and telephone conversations. The dataset contains almost 1800 documents.

Multiple versions of the TempEval corpus have also been created in different languages. Currently available languages are Arabic [73], French [88], Portuguese [89], Korean [90], Spanish [91], Romanian [92] and Italian [93]. All of the mentioned datasets are compared in Table 5.

**Table 5.** The temporal datasets in non-English languages.

Dataset	Language	Number of Tokens
ACE 2005 [66]	English, Arabic, Chinese	750,000
FR-TB [88]	French	61,000
Korean TB [90]	Korean	—
Spanish TB [91]	Spanish	68,000
IT-TimeBank [93]	Italian	150,000
TimeBank-PT [89]	Portuguese	70,000
Ro-TimeBank [92]	Romanian	65,375
Arabic TB [73]	Arabic	95,782

#### 4.8. Other Similar Corpora

The 2015 SemEval challenge featured the QA TempEval task [94], which focuses on answering temporal questions. Questions in this task were formulated in the following manner: “Is <entityA><RELATION><entityB>?” The primary objective of this task was to assess how well-recognized relations could aid in answering human questions, rather than solely evaluating their performance against official answers. Training data for this task were drawn from the annotated TempEval-3 dataset. It comprised 79 yes and no questions. The test dataset was compiled from various sources, including news articles, Wikipedia, and informational blog posts, totaling 28 documents and 294 questions. Some questions in this dataset pertained to entities mentioned more than one sentence apart in the document.

A similar task was proposed by Chen et al. [95], involving the creation of a dataset for answering time-sensitive questions. In this dataset, the task involves answering questions based on the provided text, with questions designed to necessitate a temporal understanding. For example, a question might inquire about George Washington’s position in 1777. The dataset relies on text sourced from Wikipedia articles.

## 5. Event Extraction

The construction of event knowledge graphs hinges on a fundamental task known as event extraction, which involves identifying events mentioned within a text document. In addition to recognizing events, many systems performing event extraction also aim to identify the event types and their associated attributes [96–98]. These attributes may include details about the entities participating in an event, the event’s location, timing, and more. In this section, we present and compare existing models for extracting events from text documents.

Event extraction from text has been investigated by various researchers, resulting in the development of several different methods. The scope of event extraction tasks may range from solely identifying event mentions in the document to a more comprehensive recognition of event types and their attributes.

In the context of event extraction, common terminology is used in research papers. The Oxford English Dictionary defines an event as something that happens or takes place, especially something significant or noteworthy [99]. An event mention refers to the words within a document that describe an event, with each mention typically associated with an event trigger, which is the key word that triggers the event. Researchers often define a finite set of event types under which the events of interest can be categorized. Each event type is associated with a set of expected attributes that describe the event.

While much of the work on event extraction has focused on news articles, the application of event extraction is equally important in the context of medical documents. Events in the medical domain differ significantly from those in the news domain. Medical events are often described using longer phrases, such as “a pain in the lower right arm”, whereas typical news events are described by single words or short phrases like “attack”. Furthermore, the event types relevant to the medical domain, such as “test”, “problem”, “treatment”, “clinical departments”, “evidential information”, and “clinically relevant occurrence” [69,100], differ substantially from those in news articles, which may include “attack”, “transport”, “injure”, “meet”, and more [101]. Consequently, most systems designed for event extraction from news articles may not perform well on medical documents due to these significant differences in language and event types.

### 5.1. Event Extraction Using Pattern Matching

In the early stages of automated event extraction, initial methodologies involved the creation of manually crafted rules to identify event mentions. One of the pioneering systems from this era was AutoSlog as documented in 1993 [6]. The AutoSlog system facilitated the development of event-matching patterns by suggesting potential event patterns based on pre-defined linguistic structures and a given corpus. These linguistic structures were established to represent syntactic patterns. For example, one such pattern was “<subject> active-verb”, which aimed to identify nouns acting as subjects followed by an active verb. AutoSlog utilized these linguistic patterns to generate candidate patterns for event recognition, which were then presented to users to expedite the process of crafting rules for event extraction. In an application of this system, the authors employed it to construct a model for extracting terrorist events from the MUC-4 corpus. Subsequently, the authors introduced an enhanced version of the system known as AutoSlog-TS [7]. AutoSlog-TS improved upon AutoSlog by eliminating the need for an annotated corpus. Instead, it employed the CIRCUS syntactic analyzer to automatically determine the syntactic roles of words within a corpus.

Similarly, Kim and Moldovan proposed a system called PALKA (Parallel Automatic Linguistic Knowledge Acquisition) [8], which also automatically identified event patterns from an annotated corpus. PALKA’s primary objective was to create a system that was domain-independent. Differing from the AutoSlog system, PALKA had the capability to generate patterns for recognizing multiple attributes within the context of the same event. For instance, PALKA could produce a pattern like “(perpetrator) attack (target) with (instrument)”, where the attack event would encompass three attributes: perpetrator, target, and instrument. Additionally, several other systems have been developed employing manually defined rules and patterns for event extraction and argument identification [97,102–107].

### 5.2. Event Extraction Using Traditional Machine Learning

The advent of the Automatic Content Extraction (ACE) dataset [66] marked a significant turning point in the research landscape, spurring heightened interest in the automated event extraction domain within the research community. Much of the ensuing research efforts were directed towards the development of machine learning models capable of discerning events, categorizing them by type, and identifying their associated attributes. Notably, the ACE dataset encompassed documents in three languages: English, Chinese, and Arabic. Each document containing meticulously annotated events and their corresponding attributes.

In the realm of event trigger recognition, the majority of machine learning approaches function by first classifying individual words as either event triggers or not, then classifying the remaining words to detect event attributes. Next, they recognize the attributes of recognized events and finally detect event coreference [10]. When classifying the event triggers, the trigger class is commonly subdivided into various event types [9,108]. The process is illustrated in Figure 5.

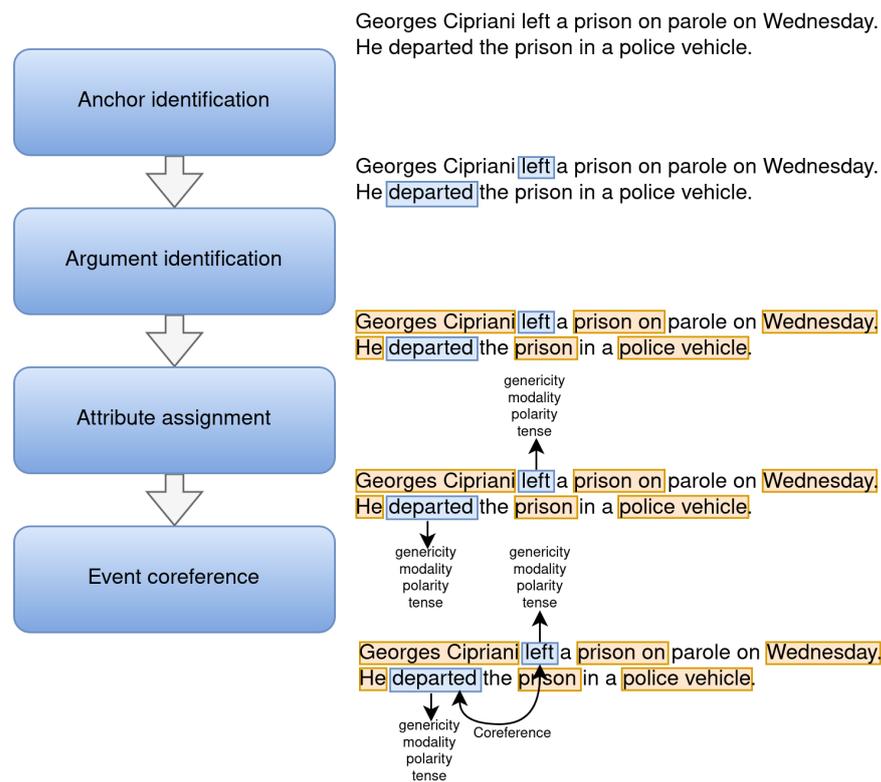


Figure 5. The process of event extraction described by Ahn [10].

The most common technique for detecting event triggers and their attributes is the use of SVM classifiers and Maximum Entropy models. Such approaches were used by several systems between the years 2010 and 2016 [48,109–112]. Yang and Mitchell (2016) [11] presented another event extraction system that relies on classical machine learning. Their approach employs conditional random fields based on manually selected features. Importantly, it does not consider each event in isolation but models relations between multiple events within the document, enhancing event-type predictions. For instance, an event of type “attack” is likely to occur in proximity to events of type “injure” and “die”, and this kind of contextual information aids in event type recognition. Similar improvements were also proposed by a number of other researchers [113–117].

### 5.3. Event Extraction Using Neural Networks

The field of event extraction has witnessed significant advancements thanks to developments in neural networks. One of the earlier neural network-based approaches was introduced by Chen et al. [12]. They proposed a system that leverages a convolutional neural network (CNN) on precomputed word embeddings to classify trigger candidates and candidate event arguments into their respective roles. The neural network takes a sentence with a marked candidate trigger or candidate argument and assigns it a role.

A notable shift in event extraction research occurred with the introduction of recurrent neural networks (RNNs) [98,118–122]. One of the earliest approaches using RNNs was proposed by Nguyen et al. [13]. They employed a Bidirectional Long Short-Term Memory (LSTM) RNN to extract events. At each token, the network identifies the event type triggered by that token and the argument role it represents concerning other potential event triggers. In 2019, Nguyen and Nguyen [49] updated this approach by replacing the LSTM layer with bidirectional gated recurrent units (GRUs). Sha et al. [14] enhanced the bidirectional LSTM approach by incorporating additional connections between words in a sentence based on their syntactic relations. Liu et al. [15] also explored the use of syntactical dependencies to enhance event recognition. Their approach involved computing token embeddings, applying a bidirectional LSTM, and then utilizing resulting vectors as input

for a graph neural network. The edges of the graph were computed based on syntactic dependencies between words. The graph neural network produced a vector representation for each word in a sentence. In the final stages, vectors representing event trigger candidates were classified to determine the event type. For recognizing event arguments, the vectors representing the event and argument candidates were combined using mean pooling and classified to ascertain the argument type. This approach yielded improved results compared to Sha et al.'s approach. Graph neural networks for event extraction have also been employed by Guo et al. [123] and Wu et al. [124]. We also identified several approaches in the medical domain using similar architectures [96,125,126].

#### 5.4. Use of Pretrained Neural Language Models

In the realm of event extraction, the adoption of pretrained language models (PLMs) has ushered in substantial advancements, with the most popular PLM for this task being Bidirectional Encoder Representations from Transformers (BERT) and its variants [127–131]. Zhang et al. [132] leveraged a BERT model to extract implicit event arguments. Their model aims to identify event arguments, including those not explicitly linked to the event mention. This is accomplished by considering BERT embeddings of the event trigger and the argument candidate head. The model employs a biaffine module to calculate the probability of a selected argument filling a particular role for the chosen event. The argument with the highest score is designated for each argument type. Following argument head determination, the network performs head-to-span expansion to identify tokens that are part of an argument based on token embeddings and the argument head. A multi-layer perceptron is employed for this task. The algorithm's evaluation was conducted on the RAMS dataset, which includes implicit arguments, making direct comparisons to algorithms designed for the ACE dataset unfeasible.

Another approach using a pretrained BERT network was proposed by [17]. Their approach involves a transition system based on token embeddings. These embeddings are derived from BERT, as well as other features, such as part-of-speech tags, character-level LSTM representations, and GloVe word embeddings. The transition system executes a sequence of actions to determine the event's position and event arguments. This methodology achieved state-of-the-art results on the ACE dataset.

In the medical domain, several systems utilize BERT models for event extraction [40,129–131]. For example, Han et al. [40] use a BERT model to perform event extraction on medical documents by classifying each token into one of the event classes. Based on the classification results, the tokens are then grouped together into events.

With the advent of large language models (LLMs), researchers have also explored using such models for tasks akin to event extraction [133–135]. These approaches typically prompt the model to annotate events within the text. Wei et al. [134] improved their predictions by employing chat-based prompts, which engage the model with a series of questions more akin to a conversation. However, despite these efforts, all observed approaches achieve performance levels significantly lower than those based on BERT [40,129–131].

#### 5.5. Cross-Lingual Event Extraction

The development of event extraction models is often hindered by the scarcity of labeled data, particularly for smaller or less-resourced languages. To address this challenge, researchers have proposed cross-lingual event extraction methods, which aim to leverage information acquired from training data in one language for event extraction in another language. Several implementations of such systems have been explored [19–21]. However, a common limitation of these approaches is their reliance on a substantial number of parallel documents between the languages, which can be challenging to obtain. Liu et al. [16] introduced a system designed to alleviate the need for a large number of parallel documents. They accomplished this by introducing context-dependent lexical mapping, which allows embeddings from one language to be translated into another. To address the issue of differing word orders between languages, they employed a convolutional graph neural

network that operates based on syntactic connections between words rather than their order within a sentence. This innovative approach was tested on both the ACE 2005 dataset and the TAC KBP 2017 dataset. In both cases, the model outperformed models trained exclusively on the target language, demonstrating its effectiveness in cross-lingual event extraction.

### 5.6. Transfer Learning

Researchers have explored innovative approaches to leverage advancements in other areas of natural language processing for event extraction tasks. One such approach was introduced by Liu et al. [22], which frames event extraction as a reading comprehension task. In this approach, the document serves as the source of knowledge for the model, and a series of questions are posed to the model about the event. Event triggers are identified using a specialized query prompt, while event attributes are extracted using prompts designed to elicit specific information, such as “What instrument did the protester use to stab an officer?” This reading comprehension-based approach offers a unique perspective on event extraction.

Another intriguing approach for event extraction was presented by Lu et al. [23]. They designed a transformer model, built upon the T5 pretrained language model, which transforms text into a structured representation. The model takes a sentence as input and outputs a structured representation containing the event, its type, and its associated arguments. This approach aims to capture the essence of events and their attributes within a structured format, offering potential benefits for downstream applications. Zhang et al. [136] explored an approach that combines text and images for event extraction. Their model connects textual and image information, combining both modalities within a single model. This multimodal approach opens up new possibilities for event extraction by incorporating visual cues from images to enhance the understanding of events described in the text.

### 5.7. Discussion

In Table 6, we compare the systems for event extraction. We compare the systems based on F1 scores on the four tasks supported by the ACE 2005 corpus. The event identification task requires the model to identify the event triggers in a document. By doing so, the model recognizes the events that are described. In the event classification task, the model has to identify the type of the detected events. Since some models solve the first two tasks in a single classification, we report only a single score in that case. The argument identification task requires the model to recognize the arguments of the identified events and the argument role classification task requires the classification of a role each argument plays in regards to the event. The presented scores are reported by each individual paper and are not directly comparable because of that; however, the reported values still give us an idea of the system’s performance. We see that the use of graph neural networks for the introduction of syntactic dependencies in a sentence proposed by Liu et al. [15] provides a big improvement. We believe that future systems for event extraction will also use graph neural networks in a similar way. Another idea that is likely to be used in future systems is the use of pretrained language models. We observed good results achieved by Zhang et al. [17,132] by using pretrained language models, so we believe that this is another promising future direction. Pretrained language models can also be combined with graph neural networks, providing benefits for both.

**Table 6.** Comparison of the described event extraction systems. We report the F1 scores for the tasks defined in the ACE 2005 challenge where available. The results are the best results reported by each of the papers.

System	Extraction Method	Corpus	Event Identification	Event Classification	Argument Identification	Argument Role Classification
AutoSlog [6]	Semi automatic pattern generation	MUC-4	-	-	-	-
PALKA [8]	Automatic pattern generation with labeled corpus	MUC-4	-	-	-	-
AutoSlog-TS [7]	Automatic pattern generation without labeled corpus	MUC-4	-	-	-	-
NYU's ACE 2005 [9]	Event extraction and entity coreference using machine learning	ACE 2005	-	-	-	-
Chen et al. (2015) [12]	Convolutional neural networks	ACE 2005	0.735	0.691	0.591	0.535
Nguyen et al. (2016) [13]	Recurrent neural networks	ACE 2005	0.719	0.693	0.628	0.554
Sha et al. (2018) [14]	Recurrent neural networks with dependency bridges	ACE 2005	-	0.719	0.677	0.587
Zhang et al. (2017) [136]	Multimodal event extraction	ACE, ERE	-	0.693	-	0.559
Zhang et al. (2019) [17]	Transition-based neural model	ACE 2005	0.761	0.738	0.574	0.533

## 6. Temporal Relation Extraction

The task of temporal relation extraction was first popularized by the creation of the TimeBank corpus [74]. The first successful approaches for recognizing temporal relations were based on hand-defined rules [24,137].

### 6.1. Temporal Relation Extraction Using Traditional Machine Learning

Temporal relation extraction, a task that involves identifying relationships between events or between events and temporal expressions, saw early efforts rooted in traditional machine learning techniques. In these approaches, machine learning models are coupled with hand-crafted features to achieve temporal relation recognition.

Mani et al. [25] tackled the recognition of temporal relation types by experimenting with various machine learning models, including Maximum Entropy (ME), Naive Bayes, and Support Vector Machine (SVM) classifiers. Their features encompass event class, aspect, modality, tense, negation, event string, and signal. Building on this foundation, Bethard [26] extended the approach to the medical domain using the TempEval 2013 dataset. Their system not only identified temporal relations but also detected events and time expressions within documents, relying on a feature set similar to that of Mani et al. [25], albeit without the signal feature.

Lin et al. [27] applied SVM models to medical documents found in the THYME and 2012 i2b2 datasets. They developed separate classifiers for recognizing relations between two events and between an event and a time expression. Features for these classifiers included part-of-speech tags, event attributes, dependency paths, and more. Ning et al. [28] further improved temporal relation prediction by introducing logical rules that considered multiple relations concurrently. For example, if event B occurred after event A and event C followed event B, then event C was inferred to be after event A. These logical rules were applied by calculating probabilities for individual relations and selecting the most likely combination. Their classification utilized a perceptron model with an expanded feature set encompassing lexical, syntactic, semantic, linguistic, and time interval features.

Table 7 provides a comparative overview of these traditional machine learning-based approaches to temporal relation extraction. Although the models were evaluated on different datasets, the commonality among them was the utilization of Support Vector Machines and Maximum Entropy models, reflecting the popularity of these traditional machine learning techniques in the domain of temporal relation extraction.

**Table 7.** Systems using hand-crafted features for temporal relation extraction. Values report accuracy for event–event (EE) relations and event–time (ET) relations.

System	Model Used	Corpus	EE	ET
Verhagen et al. (2006) [76]	Hand-crafted rules	TimeBank	-	0.64
Mani et al. (2006) [25]	SVM and ME models	TimeBank Opinion Corpus	0.625	0.761
Bethard (2013) [26]	ME model	TimeBank AQUAINT Verb-clause	0.31	-
Lin et al. (2016) [27]	SVM	THYME i2b2 2012 TimeBank	0.645	0.83
Ning et al. (2017) [28]	structured perceptron	AQUAINT Verb-clause TB Dense	0.403	-

## 6.2. Temporal Relation Extraction Using Neural Networks

In recent times, the field of temporal relation extraction has witnessed a shift towards the adoption of deep neural network architectures. These neural network-based models have proven effective in capturing complex temporal relationships in text. Table 8 presents a comparison of temporal relation extraction models employing neural networks.

Early neural network architectures in this context commonly incorporated Long Short-Term Memory (LSTM) layers on top of word embeddings to generate event embeddings, which were then used for relation classification [29,32,33].

Tourille et al. [29] introduced a deep neural network with an LSTM layer. They computed word embeddings by combining manually engineered features, word2vec embeddings, and character-level LSTM embeddings. These embeddings were then fed into a bidirectional LSTM layer. The resulting vectors from this layer were concatenated and used for classification into relation classes. This approach was tested on medical documents from the THYME corpus. Cheng and Miyao [32] adopted a similar approach but with a focus on inter-sentence temporal relation recognition. Instead of applying the LSTM over neighboring tokens, they applied it over tokens located on a syntactic dependency path connecting one event to another. This allowed them to establish connections between events described in different sentences, thereby enhancing inter-sentence temporal relation recognition.

**Table 8.** Comparison of temporal relation extraction models on several datasets. The reported F-scores are the ones achieved by the best configuration. Some models only predict the “contains” relation in a dataset and are marked with the *contains only* column. Some results are split into scores for event–event (EE) relations and event–time (ET) relations.

System	Model	Contains Only	F-Score
Dataset: THYME			
Tourille et al. (2017) [29]	Bi-LSTM	✗	0.683
Dligach et al. (2017) [30]	CNN	✓	EE: 0.54, ET: 0.71
Lin et al. (2019) [31]	BERT	✓	0.684
Dataset: TimeBank-Dense			
Cheng and Miyao (2017) [32]	Bi-LSTM	✗	EE: 0.53, ET: 0.47
Leeuwenberg and Moens (2018) [33]	Bi-LSTM	✗	0.561
Zhou et al. (2021) [34]	soft logic	✗	0.652
Zhang et al. (2021) [35]	BERT + GNN	✗	0.667
Xu et al. (2021) [36]	BERT + GNN	✗	0.732
Mathur et al. (2021) [37]	BERT + GNN	✗	0.678
Yuan et al. (2023) [138]	ChatGPT	✗	0.366
Chan et al. (2023) [139]	ChatGPT	✗	0.233
Dataset: i2b2 2012			
Zhou et al. (2021) [34]	Soft logic	✗	0.802
Dataset: MATRES			
Zhang et al. (2021) [35]	BERT + GNN	✗	0.793
Mathur et al. (2021) [37]	BERT + GNN	✗	0.823
Yuan et al. (2023) [138]	ChatGPT	✗	0.193
Chan et al. (2023) [139]	ChatGPT	✗	0.350

Dligach et al. [30] delved deeper into the use of neural networks for temporal relation extraction. They explored various network architectures and found that neural networks outperformed hand-engineered features. Interestingly, they discovered that convolutional neural network (CNN) architectures performed better than LSTM networks. Their experiments were conducted on the THYME dataset, which primarily focused on the “contains” relation (referred to as “during” in Alen’s categorization in Figure 4).

In addition to LSTM and CNN architectures, researchers have also explored the application of other neural network architectures, such as graph neural networks [140] and attention-based neural networks [141]. These architectures offer diverse approaches to capturing temporal relations in text, contributing to the advancement of the field.

### 6.3. Use of Pretrained Neural Language Models

The recent breakthroughs in natural language processing have been driven by the widespread use of large pretrained language models. These models, trained on vast amounts of text, leverage their learned knowledge to excel in various language-related tasks, including temporal relation extraction [28,31,34,142].

Lin et al. [31] introduced an early approach to temporal relation extraction using pretrained language models. Their method involved marking events with special tokens and processing the sequence using a BERT network. The final classification relied on the embedding of a “[CLS]” token. This model primarily recognized “CONTAINS” and “CONTAINED\_BY” relations and experimented with BERT models pretrained on different types of text, with models pretrained on more relevant data, yielding improved results. Zhou et al. [34] built on top of this approach by introducing probabilistic soft logic, incorporating logical rules that must hold between temporal relations. This approach, similar to the work by Ning et al. [28], significantly improved results over the baseline BERT model. Their evaluation encompassed the 2012 i2b2 medical dataset and the TimeBank-Dense dataset.

To improve on simply classifying temporal relations based on embeddings provided by BERT, some researchers proposed to complement them by graph neural networks (GNNs), which offer additional information about sentence structure and inter-sentence relations. Zhang et al. [35] employed BERT to compute token embeddings and then used two graph neural networks—one that accepted events and related tokens and another that accepted the shortest syntactic path between events. This approach yielded a 5% improvement compared to only using BERT when tested on the TimeBank-Dense and MATRES datasets. Xu et al. [36] proposed another approach, taking advantage of a GNN to improve relation extraction performance. The approach incorporated attention mechanisms, outperforming the LSTM-based architecture when evaluated on the TimeBank-Dense dataset. Mathur et al. [37] extended the use of graph neural networks in conjunction with BERT embeddings for document-level temporal relation extraction. Their approach incorporated multiple graph neural networks with gating convolutional layers, achieving superior results compared to baselines across multiple datasets, including MATRES, TimeBank-Dense, and a subset of the TDDiscourse corpus.

#### 6.4. Use of External Knowledge

When humans comprehend natural language, they rely on their shared knowledge to make educated guesses about implicit details not explicitly mentioned in the text. For instance, if a text describes someone as feeling hungry and having lunch, we intuitively understand that the hunger likely preceded the meal and that the meal itself probably took a relatively short amount of time. These inferences are drawn from our common background knowledge and contextual cues.

In contrast, computer models often lack access to this common knowledge and can only extract information explicitly stated in the text. To address this limitation, researchers have proposed methods to integrate general knowledge into computer models [38–40,143].

One approach to incorporating general knowledge into models involves compiling statistics about target variables from a large corpus of text. For instance, by observing temporal relations between events in a corpus, we can predict the likelihood of each relation occurring between the observed events. This statistical knowledge aids the model in relation extraction. Ning et al. [38] developed such a statistical resource, containing probabilities for temporal relations between events in the news domain. This resource was constructed using news articles from The New York Times over a 20-year period. Their findings demonstrate that integrating this resource enhances the performance of existing temporal relation extraction systems. In subsequent work, Ning et al. [39] introduced a novel state-of-the-art system for temporal relation extraction that leverages this statistical resource. Their model employs an architecture that applies a Long Short-Term Memory (LSTM) network over contextualized token embeddings. Additional knowledge is incorporated through a common sense encoder, which is concatenated with the event token embeddings. This inclusion of common sense knowledge led to a 3% improvement in results. Han et al. [40] proposed a similar approach for temporal relation classification that also integrates common sense knowledge. Their model begins by computing BERT token embeddings, which are then passed through an LSTM layer. The resulting representation is used to extract events and predict probabilities for their temporal relations. Events are then associated with their types, and probabilities of temporal relations between them are estimated based on corpus statistics. The relation probabilities predicted by the model are combined with those computed from corpus statistics using maximum a posteriori estimation. This approach employs probabilistic constraints instead of rigid constraints.

Certain authors also utilize constraints grounded in their domain knowledge for temporal relation extraction. Zhou et al. [34] and Ning et al. [28] applied transitivity between temporal relations to enhance their predictions. These types of constraints introduce domain-specific knowledge into the model, even without statistics from a large corpus.

### 6.5. Large Language Models

With the rise of large language models like ChatGPT and GPT-4 [144], which have demonstrated their effectiveness across various tasks, researchers have explored the possibility of using such general models for classification tasks like temporal relation extraction.

The primary advantage of these approaches is that they enable zero-shot classification [138,139,145]. This means that these models can be used to extract temporal relations without requiring any specific training data that provide examples of temporal relations. Researchers have investigated the use of such models by fine-tuning or adapting prompts to optimize extraction reliability. However, it is worth noting that the prediction accuracy of these general models tends to be considerably lower than that of purpose-built models [138,139].

To enhance the performance of these models, efforts have been made to design prompts that provide explanations about the question setting and causal relations [139]. Yuan et al. [138] took a step further by introducing "chain of thought" prompts, where the model is prompted with a series of questions concerning the timing of events. Despite these improvements, researchers still report significantly lower performance compared to some approaches utilizing other specialized architectures for temporal relation extraction.

### 6.6. Discussion

Temporal relations are the most commonly extracted temporal features of events in the papers that we analyzed. In order to compare the systems for temporal relation extraction, we present them in two separate tables. Table 7 compares systems using traditional machine learning models, and Table 8 compares systems using neural network architectures. Since the models using traditional machine learning use different datasets, we cannot make meaningful comparisons of their performance. Recent research suggests that the use of language models like BERT and graph neural networks provides the best results [35–37].

We also believe that the use of common knowledge resources could provide large benefits in the process of extracting temporal relations. Ning et al. [39] and Han et al. [40] showed significant improvements to the results when using statistical resources to complement extracted relations. We believe that there is still a lot of unexplored potential for the use of common sense knowledge in the process of temporal relation extraction. Research in other areas of natural language processing has shown that the integration of common sense knowledge is an effective way of improving model performance, especially in cases where not a lot of training data are available. Liu et al. [146] demonstrated the improvements that come from additional domain knowledge on a variety of natural language tasks.

## 7. Timeline and Knowledge Graph Construction

Once information about events and their temporal attributes is extracted from text, the next step is to represent this information in a machine-readable format. A common representation for knowledge extracted from plain text documents is a knowledge graph [1]. Most existing knowledge graphs are centered around entities. In such graphs, nodes represent entities, and edges represent relations between them. However, we can also use knowledge graphs to represent event information, leading to event-centric knowledge graphs.

Event-centric knowledge graphs, like EventKG constructed by Gottschalk [147], focus on events as the central entities. EventKG was created by extracting events from various sources, including Wikidata [2], DBpedia [1], YAGO [3], Wikipedia Event Lists, and Wikipedia Current Events Portal. It comprises 322,669 events with 88,473,111 relations, including information about the time and location of events when available.

In this survey, we explore the automatic generation of event-centric knowledge graphs from unstructured documents. Automatically constructing such graphs can have various applications, including event information analysis and using the extracted knowledge graphs as sources of external knowledge as discussed in Section 6.4. To automatically create event-centric knowledge graphs, we need to integrate the methods for informa-

tion extraction described earlier with techniques for disambiguation and constructing knowledge graphs.

### 7.1. Constructing Temporal Knowledge Graphs

Adding temporal information to a knowledge graph to create a temporal knowledge graph has gained significant attention in recent years [148]. One example of automatically building an event-centric temporal knowledge graph was presented by Rospocher et al. [46]. They developed a comprehensive pipeline of existing natural language processing tools capable of extracting information from plain text news articles to construct a knowledge graph containing events and their associated attributes. Another system for constructing event-centric knowledge graphs was introduced by Song [149]. This system crawls news articles from the web to extract credit events and represent them as nodes and their causal relations as edges in a knowledge graph. The system detects events and their causal relations based on syntactic dependencies between words. These methods rely on traditional techniques for extracting events and temporal relations and could potentially be enhanced with the integration of deep learning methods.

Similar approaches have been employed to construct knowledge graphs, such as the Global Database of Event Language and Tone (GDELT) [150] and the Integrated Crisis Early Warning System (ICEWS) [151]. These knowledge graphs contain temporal information about events sourced from news articles, government reports, and other publicly available documents. The methods used in these cases rely on statistical natural language processing techniques, like Dirichlet Localization Algorithms (ALDs). EventKG [147] is another example of a temporal knowledge graph. EventKG is constructed from existing structured data sources, including DBpedia, Wikidata, Yago, and Wikipedia. Unlike the previously mentioned models, EventKG does not work directly with natural language text but starts with structured data. The construction of EventKG involves several steps, including identifying events, extracting relations, integrating events and entities, and fusing the temporal and spatial information associated with the events.

### 7.2. Estimating Event Timelines

The representation of temporal information about events in the form of timelines can provide a more intuitive understanding for humans. Timelines can capture the sequencing of events and their durations, making them easier to comprehend. However, while timelines are beneficial for human interpretation, they may lack the ability to record additional extracted data compared to knowledge graphs. Timelines can be categorized as relative or absolute. Relative timelines focus on capturing the sequence of events without specifying their actual time points, relying on temporal relations between events. In contrast, absolute timelines record the actual times of events, enabling the combination of multiple timelines. Absolute timelines are more informative but require more extensive temporal information. Leeuwenberg and Moens [33] proposed an architecture to create relative event timelines directly. Instead of predicting temporal relations, they predict relative times, allowing the direct construction of timelines. Their model predicts relative start and end times for events, where greater values indicate later times. The architecture employs two Bidirectional Long Short-Term Memory (Bi-LSTM) layers, one for predicting event start times and another for predicting event durations. These predictions are combined with token embeddings of event and time expressions. Evaluation is performed on the THYME medical corpus by checking if the predicted time points align with the annotated relations. Similarly, Vashishtha et al. [42] constructed a dataset containing relative timeline positions of events in a document. Their model takes pairs of events within a sentence and positions them relative to each other on a timeline. Subsequently, these relative timelines are combined into a single document timeline. The model also predicts the duration class (e.g., seconds, minutes, hours, and days) for events in the timeline. In 2020, Leeuwenberg and Moens [41] extended their work to extract absolute event times and durations. They annotated the i2b2 medical corpus with the event starting times, ending times, and durations. The model for

predicting absolute event times uses the ELMo embeddings of words, which are processed through an LSTM or a convolutional layer. The predictions are generated using a multi-layer perceptron. For predicting event durations, the model considers only the event and its surrounding context. The predicted values represent the absolute time of the event and its duration.

### 7.3. Estimating Event Duration

Predicting the duration of events is a crucial aspect of temporal information extraction. Various approaches have been explored for this task, ranging from binary classification to fine-grained classification and continuous variable prediction. Pan et al. [43] created a corpus of news articles and annotated events from the TimeBank corpus with their durations. They designed a binary classification task to predict whether events are short (less than a day) or long (more than a day), using hand-crafted features, including event properties, bag-of-words vectors, subject and object of the events, and WordNet hypernyms. Support Vector Machine (SVM) achieved the best results for this binary classification. Gusev et al. [44] extended Pan et al.'s work by expanding the dataset with additional annotations and predicting multiple duration classes, such as seconds, minutes, hours, etc. They employed a similar set of features, including event attributes, named entity classes, typed dependencies, and whether an event is a reporting verb. Various machine learning models, including Naive Bayes, Logistic Regression, Maximum Entropy, and SVM, were evaluated, with Maximum Entropy achieving the best performance. Additionally, Gusev et al. proposed an unsupervised model to extract event durations using patterns. Vempala et al. [45] introduced a neural network-based approach for event duration estimation. Their model utilizes three Long Short-Term Memory (LSTM) networks over word embeddings in a sentence: one capturing information from the entire sentence, one from words before the event, and one from words after the event. These vectors, combined with the event's embedding, are classified using a softmax layer. While they improved upon the previous binary classification models, their approach still predicted event duration as a binary output.

Several models designed for event timeline construction also incorporate event duration prediction, which is crucial for the timeline sequencing task [33,41,42]. Vashishtha et al. [42] performed the fine-grained classification of event durations, classifying events into categories such as seconds, minutes, hours, days, weeks, months, years, decades, centuries, and forever.

### 7.4. Discussion

Presenting extracted information in a knowledge graph is important for increasing its usability. Most of the research aiming to create a structured representation of events focuses on creating event timelines. While these can be useful, we suggest that future projects present the extracted data in the form of a knowledge graph, as these can capture more diverse information. Once the necessary information has been captured in a knowledge graph, a timeline can be constructed as an additional visualization of the events.

The systems constructing event timelines are also limited since many of them only provide relative timelines [33,42]. We propose that future work focuses on anchoring the timelines to absolute times, following the example of Leeuwenberg and Moens [41]. Absolute times are more useful since they can be directly combined with events extracted from other documents. A challenge when predicting the absolute times of events is that dates and times of the events are often not mentioned in a document. This can be solved by using the document creation time as a reference point.

## 8. Discussion and Future Research Directions

In this section, we discuss how to combine the existing technologies into a single system for constructing event-centric knowledge graphs from unstructured documents. We describe how the challenges have been solved so far and which remain unsolved.

Incorporating common knowledge into event-centric knowledge graph construction can significantly enhance the accuracy of key tasks. Utilizing domain-specific common knowledge and statistics can be valuable for improving tasks such as temporal relation prediction and event duration estimation. Researchers may consider implementing a mechanism to automatically generate and update common knowledge collections based on the information extracted by the algorithm to ensure ongoing improvement. This is especially relevant for the domain of temporal knowledge graphs, as the temporal patterns that occurred in previously analyzed documents are likely to repeat in the future and can thus help a lot with extracting new temporal information.

Developing end-to-end systems for event knowledge graph construction is an essential goal. Such systems can reduce error propagation and enhance overall efficiency by combining multiple stages of the knowledge graph construction process. The creation of a unified architecture that can perform event extraction, the recognition of temporal properties, and knowledge graph construction tasks within a single model would streamline the process and reduce potential errors. While such systems are beneficial in all knowledge graph construction applications, they are especially relevant for constructing knowledge graphs focused on temporal information, as such construction generally requires pipelines with many stages.

Future research should expand its focus to include the automatic creation of event-centric knowledge graphs in addition to existing event timelines. These knowledge graphs can provide a more comprehensive representation of events, including details such as participants, location, and event type. To ensure interoperability between different data sources and applications, utilizing standardized knowledge graph schemas and defining a common representation for event-centric knowledge graphs is crucial.

The prediction of absolute temporal attributes, including event start times, end times, and durations, remains an underexplored area. Addressing this gap requires the creation of manually annotated datasets that include this information to effectively train and evaluate models. In cases where manual annotation is limited, researchers may explore the use of semi-automated annotations, combining hand-annotated examples with a substantial number of silver annotations generated using ensemble algorithms to expand training data while mitigating potential errors. This approach will facilitate the recognition of absolute event times and enhance the capabilities of knowledge graph construction systems.

Integrating temporal knowledge graphs into question-answering (Q&A) systems also presents an intriguing avenue for research. This involves devising methods that enable the extraction of temporal context from queries and leveraging it to provide temporally aware and accurate answers. In essence, advancements in time representation, time relationship extraction, time reasoning, and Q&A are interconnected, as they collectively contribute to the construction of more dynamic and contextually aware temporal knowledge graphs.

Constructing temporal knowledge graphs also necessitates a critical consideration of ethical dimensions and potential biases inherent in the data sources and algorithms involved. The selection and curation of data may inadvertently introduce biases, reflecting historical inequalities or skewed perspectives. Ethical concerns arise in determining whose narratives are prioritized and how events are framed within a temporal context. Biases can also emerge through algorithmic decisions, impacting the representation of certain events or communities. Transparency and inclusivity are paramount, demanding careful scrutiny of the methods employed to construct these graphs. Vigilance is essential to mitigate ethical pitfalls, ensuring that temporal knowledge graphs not only accurately reflect temporal relationships but also adhere to principles of fairness, inclusiveness, and ethical data use. Regular audits and ongoing ethical considerations should be integral to the construction and maintenance of these graphs to foster a responsible and unbiased foundation for knowledge representation.

## 9. Conclusions

Event-centric temporal knowledge graphs are useful in many real-world applications. Such applications are event-centric question answering and timeline generation [147]. By incorporating data from specific domains into such knowledge graphs, we can enable even more applications like, for example, cross-cultural and cross-lingual event-centric analytics [152,153], which can be enabled by including events collected from documents from multiple cultural backgrounds.

In the field of medical research, the construction of event-centric temporal graphs can be used to extract information from a large number of unstructured medical notes. Such information can then be used to analyze patterns and trends across many patients to solve real-world tasks, such as disease trajectory detection [154] and clinical dead-end prediction [155].

In this survey, we delved into the existing methodologies and datasets for constructing event-centric temporal knowledge graphs from unstructured documents. We structured the process into three main stages: event extraction, the extraction of temporal attributes and relations, and knowledge graph construction. Additionally, we examined the common schemes for representing event information, temporal relations, and time properties, while also highlighting the prevalence of pretrained language models in current systems.

Our findings indicate that models employing pretrained language models, particularly BERT, achieved remarkable success in the extraction of events, temporal attributes, and temporal relations. Furthermore, domain-specific pretrained models demonstrated even higher effectiveness. However, the task of seamlessly integrating these extracted data into event-centric knowledge graphs remains an understudied area. We also find that the pretrained conversational models like ChatGPT perform poorly when applied directly to most of the tasks connected to event-centric knowledge graph construction.

Building on our analysis, we propose several promising directions for future research. Leveraging common knowledge in information extraction processes could enhance results, especially in domains with limited labeled data. Additionally, there is a pressing need to develop techniques for extracting absolute temporal attributes, as most current systems primarily focus on relative event times. The creation of automated event-centric knowledge graphs and end-to-end systems capable of executing all of the aforementioned tasks should also be prioritized in future research endeavors. This way, we can move closer to fully automating the construction of event-centric temporal knowledge graphs from unstructured documents.

**Author Contributions:** Investigation, T.K.; writing—original draft preparation, T.K.; writing—review and editing, S.Ž.; visualization, T.K.; supervision, S.Ž.; project administration, S.Ž. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Slovenian Research Agency in the Young Researchers grant.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** No new data were created or analyzed in this study. Data sharing is not applicable to this article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P.N.; Hellmann, S.; Morsey, M.; Van Kleef, P.; Auer, S.; et al. Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia. *Semant. Web* **2015**, *6*, 167–195. [CrossRef]
2. Vrandečić, D.; Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]
3. Suchanek, F.M.; Kasneci, G.; Weikum, G. Yago: A core of semantic knowledge. In Proceedings of the 16th International Conference on World Wide Web, Banff, AB, Canada, 8–12 May 2007; pp. 697–706.

4. Wang, Y.; Zhu, M.; Qu, L.; Spaniol, M.; Weikum, G. Timely yago: Harvesting, querying, and visualizing temporal knowledge from wikipedia. In Proceedings of the 13th International Conference on Extending Database Technology, Lausanne, Switzerland, 22–26 March 2010; pp. 697–700.
5. Hwang, J.D.; Bhagavatula, C.; Le Bras, R.; Da, J.; Sakaguchi, K.; Bosselut, A.; Choi, Y. (Comet-) atomic 2020: On symbolic and neural commonsense knowledge graphs. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; Volume 35, pp. 6384–6392.
6. Riloff, E. Automatically constructing a dictionary for information extraction tasks. In Proceedings of the Eleventh National Conference on Artificial Intelligence, Washington, DC, USA, 11–15 July 1993; pp. 811–816.
7. Riloff, E.; Shoen, J. Automatically acquiring conceptual patterns without an annotated corpus. In Proceedings of the Third Workshop on Very Large Corpora, Cambridge, MA, USA, 30 June 1995.
8. Kim, J.T.; Moldovan, D.I. Acquisition of linguistic patterns for knowledge-based information extraction. *IEEE Trans. Knowl. Data Eng.* **1995**, *7*, 713–724.
9. Grishman, R.; Westbrook, D.; Meyers, A. Nyu’s english ace 2005 system description. *ACE* **2005**, *5*, 2.
10. Ahn, D. The stages of event extraction. In Proceedings of the Workshop on Annotating and Reasoning about Time and Events, Sydney, Australia, 23 July 2006; pp. 1–8.
11. Yang, B.; Mitchell, T. Joint Extraction of Events and Entities within a Document Context. In Proceedings of the NAACL-HLT, San Diego, CA, USA, 17 June 2016; pp. 289–299.
12. Chen, Y.; Xu, L.; Liu, K.; Zeng, D.; Zhao, J. Event extraction via dynamic multi-pooling convolutional neural networks. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, 26–31 July 2015; pp. 167–176.
13. Nguyen, T.H.; Cho, K.; Grishman, R. Joint event extraction via recurrent neural networks. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 300–309.
14. Sha, L.; Qian, F.; Chang, B.; Sui, Z. Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
15. Liu, X.; Luo, Z.; Huang, H. Jointly multiple events extraction via attention-based graph information aggregation. *arXiv* **2018**, arXiv:1809.09078.
16. Liu, J.; Chen, Y.; Liu, K.; Zhao, J. Neural cross-lingual event detection with minimal parallel resources. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 738–748.
17. Zhang, J.; Qin, Y.; Zhang, Y.; Liu, M.; Ji, D. Extracting Entities and Events as a Single Task Using a Transition-Based Neural Model. In Proceedings of the IJCAI, Macao, China, 10–16 August 2019; pp. 5422–5428.
18. Zhang, J.; Liu, M.; Zhang, Y. Topic-informed neural approach for biomedical event extraction. *Artif. Intell. Med.* **2020**, *103*, 101783. [[CrossRef](#)]
19. Ji, H. Cross-lingual predicate cluster acquisition to improve bilingual event extraction by inductive learning. In Proceedings of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics, Boulder, CO, USA, 5 June 2009; pp. 27–35.
20. Zhu, Z.; Li, S.; Zhou, G.; Xia, R. Bilingual event extraction: A case study on trigger type determination. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Baltimore, MD, USA, 22–27 June 2014; pp. 842–847.
21. Chen, Z.; Ji, H. Can one language bootstrap the other: A case study on event extraction. In Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing, Boulder, CO, USA, 4 June 2009; pp. 66–74.
22. Liu, J.; Chen, Y.; Liu, K.; Bi, W.; Liu, X. Event extraction as machine reading comprehension. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 1641–1651.
23. Lu, Y.; Lin, H.; Xu, J.; Han, X.; Tang, J.; Li, A.; Sun, L.; Liao, M.; Chen, S. Text2Event: Controllable Sequence-to-Structure Generation for End-to-end Event Extraction. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 16–20 November 2021; pp. 2795–2806.
24. Gaizauskas, R.; Harkema, H.; Hepple, M.; Setzer, A. Task-oriented extraction of temporal information: The case of clinical narratives. In Proceedings of the Thirteenth International Symposium On Temporal Representation And Reasoning (time’06), Budapest, Hungary, 15–17 June 2006; pp. 188–195.
25. Mani, I.; Verhagen, M.; Wellner, B.; Lee, C.; Pustejovsky, J. Machine learning of temporal relations. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, 17–18 July 2006; pp. 753–760.
26. Bethard, S. Cleartk-timeml: A minimalist approach to tempeval 2013. In Proceedings of the Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, GA, USA, 14–15 June 2013; pp. 10–14.

27. Lin, C.; Dligach, D.; Miller, T.A.; Bethard, S.; Savova, G.K. Multilayered temporal modeling for the clinical domain. *J. Am. Med. Inform. Assoc.* **2016**, *23*, 387–395. [[CrossRef](#)] [[PubMed](#)]
28. Ning, Q.; Feng, Z.; Roth, D. A Structured Learning Approach to Temporal Relation Extraction. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; pp. 1027–1037.
29. Tourille, J.; Ferret, O.; Neveol, A.; Tannier, X. Neural architecture for temporal relation extraction: A Bi-LSTM approach for detecting narrative containers. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 224–230.
30. Dligach, D.; Miller, T.; Lin, C.; Bethard, S.; Savova, G. Neural temporal relation extraction. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, Valencia, Spain, 3–7 April 2017; pp. 746–751.
31. Lin, C.; Miller, T.; Dligach, D.; Bethard, S.; Savova, G. 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, Minneapolis, MN, USA, 7 June 2019; pp. 65–71.
32. Cheng, F.; Miyao, Y. Classifying temporal relations by bidirectional LSTM over dependency paths. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1–6.
33. Leeuwenberg, A.; Moens, M.F. Temporal Information Extraction by Predicting Relative Time-lines. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 1237–1246.
34. Zhou, Y.; Yan, Y.; Han, R.; Caufield, H.J.; Chang, K.W.; Sun, Y.; Ping, P.; Wang, W. Clinical Temporal Relation Extraction with Probabilistic Soft Logic Regularization and Global Inference. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'21), Virtual, 2–9 February 2021.
35. Zhang, S.; Huang, L.; Ning, Q. Extracting Temporal Event Relation with Syntactic-Guided Temporal Graph Transformer. *arXiv* **2021**, arXiv:2104.09570.
36. Xu, X.; Gao, T.; Wang, Y.; Xuan, X. Event temporal relation extraction with attention mechanism and graph neural network. *Tsinghua Sci. Technol.* **2021**, *27*, 79–90. [[CrossRef](#)]
37. Mathur, P.; Jain, R.; Derroncourt, F.; Morariu, V.; Tran, Q.H.; Manocha, D. TIMERS: Document-level Temporal Relation Extraction. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Virtual, 1–6 August 2021; pp. 524–533.
38. Ning, Q.; Wu, H.; Peng, H.; Roth, D. Improving Temporal Relation Extraction with a Globally Acquired Statistical Resource. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 841–851.
39. Ning, Q.; Subramanian, S.; Roth, D. An Improved Neural Baseline for Temporal Relation Extraction. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 6203–6209.
40. Han, R.; Zhou, Y.; Peng, N. Domain Knowledge Empowered Structured Neural Net for End-to-End Event Temporal Relation Extraction. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 5717–5729.
41. Leeuwenberg, A.; Moens, M.F. Towards extracting absolute event timelines from english clinical reports. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 2710–2719. [[CrossRef](#)]
42. Vashishtha, S.; Van Durme, B.; White, A.S. Fine-grained temporal relation extraction. *arXiv* **2019**, arXiv:1902.01390.
43. Pan, F.; Mulkar-Mehta, R.; Hobbs, J.R. Annotating and learning event durations in text. *Comput. Linguist.* **2011**, *37*, 727–752. [[CrossRef](#)]
44. Gusev, A.; Chambers, N.; Khilnani, D.R.; Khaitan, P.; Bethard, S.; Jurafsky, D. Using query patterns to learn the duration of events. In Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011), Oxford, UK, 12–14 January 2011.
45. Vempala, A.; Blanco, E.; Palmer, A. Determining event durations: Models and error analysis. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 164–168.
46. Rospocher, M.; van Erp, M.; Vossen, P.; Fokkens, A.; Aldabe, I.; Rigau, G.; Soroa, A.; Ploeger, T.; Bogaard, T. Building event-centric knowledge graphs from news. *J. Web Semant.* **2016**, *37*, 132–151. [[CrossRef](#)]
47. Ma, M.D.; Sun, J.; Yang, M.; Huang, K.H.; Wen, N.; Singh, S.; Han, R.; Peng, N. Eventplus: A temporal event understanding pipeline. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations, Online, 6–11 June 2021; pp. 56–65.
48. Jindal, P.; Roth, D. Extraction of events and temporal expressions from clinical narratives. *J. Biomed. Inform.* **2013**, *46*, S13–S19. [[CrossRef](#)] [[PubMed](#)]
49. Nguyen, T.M.; Nguyen, T.H. One for all: Neural joint modeling of entities and events. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 6851–6858.

50. Han, X.; Gao, T.; Lin, Y.; Peng, H.; Yang, Y.; Xiao, C.; Liu, Z.; Li, P.; Zhou, J.; Sun, M. More Data, More Relations, More Context and More Openness: A Review and Outlook for Relation Extraction. In Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, Suzhou, China, 4–7 December 2020; pp. 745–758.
51. Kumar, S. A survey of deep learning methods for relation extraction. *arXiv* **2017**, arXiv:1705.03645.
52. Pawar, S.; Palshikar, G.K.; Bhattacharyya, P. Relation extraction: A survey. *arXiv* **2017**, arXiv:1712.05191.
53. Smirnova, A.; Cudré-Mauroux, P. Relation extraction using distant supervision: A survey. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–35. [[CrossRef](#)]
54. Liu, K. A survey on neural relation extraction. *Sci. China Technol. Sci.* **2020**, *63*, 1971–1989. [[CrossRef](#)]
55. Bose, P.; Srinivasan, S.; Sleeman IV, W.C.; Palta, J.; Kapoor, R.; Ghosh, P. A survey on recent named entity recognition and relationship extraction techniques on clinical texts. *Appl. Sci.* **2021**, *11*, 8319. [[CrossRef](#)]
56. Xiang, W.; Wang, B. A survey of event extraction from text. *IEEE Access* **2019**, *7*, 173111–173137. [[CrossRef](#)]
57. Gumiel, Y.B.; Silva e Oliveira, L.E.; Claveau, V.; Grabar, N.; Paraiso, E.C.; Moro, C.; Carvalho, D.R. Temporal Relation Extraction in Clinical Texts: A Systematic Review. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–36. [[CrossRef](#)]
58. Alfattni, G.; Peek, N.; Nenadic, G. Extraction of temporal relations from clinical free text: A systematic review of current approaches. *J. Biomed. Inform.* **2020**, *108*, 103488. [[CrossRef](#)] [[PubMed](#)]
59. Abu-Salih, B. Domain-specific knowledge graphs: A survey. *J. Netw. Comput. Appl.* **2021**, *185*, 103076. [[CrossRef](#)]
60. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Philip, S.Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 494–514. [[CrossRef](#)] [[PubMed](#)]
61. Yan, Z.; Tang, X. Hierarchical storyline generation based on event-centric temporal knowledge graph. In Proceedings of the International Symposium on Knowledge and Systems Sciences, Beijing, China, 11–12 June 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 149–159.
62. Wang, X.; Liu, K.; Wang, D.; Wu, L.; Fu, Y.; Xie, X. Multi-level recommendation reasoning over knowledge graphs with reinforcement learning. In Proceedings of the ACM Web Conference 2022, Lyon, France, 25–29 April 2022; pp. 2098–2108.
63. Bonifazi, G.; Cecchini, S.; Corradini, E.; Giuliani, L.; Ursino, D.; Virgili, L. Extracting time patterns from the lifespans of TikTok challenges to characterize non-dangerous and dangerous ones. *Soc. Netw. Anal. Min.* **2022**, *12*, 62. [[CrossRef](#)]
64. Bonifazi, G.; Cauteruccio, F.; Corradini, E.; Marchetti, M.; Sciarretta, L.; Ursino, D.; Virgili, L. A Space-Time Framework for Sentiment Scope Analysis in Social Media. *Big Data Cogn. Comput.* **2022**, *6*, 130. [[CrossRef](#)]
65. Pustejovsky, J.; Sauri, R.; Setzer, A.; Gaizauskas, R.; Ingria, B. *TimeML Annotation Guidelines*; Brandeis University: Waltham, MA, USA, 2002; Volume 23.
66. Walker, C.; Strassel, S.; Medero, J.; Maeda, K. *ACE 2005 Multilingual Training Corpus*; Linguistic Data Consortium: Philadelphia, PA, USA, 2005.
67. Boguraev, B.; Pustejovsky, J.; Ando, R.; Verhagen, M. TimeBank evolution as a community resource for TimeML parsing. *Lang. Resour. Eval.* **2007**, *41*, 91–115. [[CrossRef](#)]
68. Graff, D. *The AQUAINT Corpus of English News Text*; Linguistic Data Consortium: Philadelphia, PA, USA, 2002.
69. Sun, W.; Rumshisky, A.; Uzuner, O. Evaluating temporal relations in clinical text: 2012 i2b2 Challenge. *J. Am. Med. Inform. Assoc.* **2013**, *20*, 806–813. [[CrossRef](#)]
70. Mazur, P.; Dale, R. Wikiwars: A new corpus for research on temporal expressions. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, USA, 9–11 October 2010; pp. 913–922.
71. Allen, J.F.; Hayes, P.J. Moments and points in an interval-based temporal logic. *Comput. Intell.* **1989**, *5*, 225–238. [[CrossRef](#)]
72. Sauri, R.; Littman, J.; Knippen, B.; Gaizauskas, R.; Setzer, A.; Pustejovsky, J. TimeML annotation guidelines. *Version* **2006**, *1*, 31.
73. Haffar, N.; Hkiri, E.; Zrigui, M. TimeML Annotation of Events and Temporal Expressions in Arabic Texts. In Proceedings of the Computational Collective Intelligence, Hendaye, France, 4–6 September 2019; Nguyen, N.T., Chbeir, R., Exposito, E., Aniórté, P., Trawiński, B., Eds.; Springer: Cham, Switzerland, 2019; pp. 207–218.
74. Pustejovsky, J.; Hanks, P.; Sauri, R.; See, A.; Gaizauskas, R.; Setzer, A.; Radev, D.; Sundheim, B.; Day, D.; Ferro, L.; et al. The timebank corpus. In Proceedings of the Corpus Linguistics, Lancaster, UK, 28–31 March 2003; Volume 2003, p. 40.
75. Pustejovsky, J.; Littman, J.; Sauri, R.; Verhagen, M. *Timebank 1.2 Documentation*; Linguistic Data Consortium: Philadelphia, PA, USA, 2006; pp. 6–11; Event London no. April
76. Verhagen, M.; Gaizauskas, R.; Schilder, F.; Hepple, M.; Katz, G.; Pustejovsky, J. Semeval-2007 task 15: Tempeval temporal relation identification. In Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic, 23–24 June 2007; pp. 75–80.
77. Verhagen, M.; Sauri, R.; Caselli, T.; Pustejovsky, J. SemEval-2010 Task 13: TempEval-2. In Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, 15–16 July 2010; pp. 57–62.
78. UzZaman, N.; Llorens, H.; Derczynski, L.; Allen, J.; Verhagen, M.; Pustejovsky, J. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In Proceedings of the Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, GA, USA, 14–15 June 2013; pp. 1–9.
79. Parker, R.; Graff, D.; Kong, J.; Chen, K.; Maeda, K. *English Gigaword*, 5th ed.; Linguistic Data Consortium: Philadelphia, PA, USA, 2011.

80. Cassidy, T.; McDowell, B.; Chambers, N.; Bethard, S. *An Annotation Framework for Dense Event Ordering*; Technical Report; Carnegie-Mellon University: Pittsburgh, PA, USA, 2014.
81. Zhong, X.; Sun, A.; Cambria, E. Time expression analysis and recognition using syntactic token types and general heuristic rules. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 420–429.
82. Bethard, S.; Derczynski, L.; Savova, G.; Pustejovsky, J.; Verhagen, M. Semeval-2015 task 6: Clinical tempeval. In Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, CO, USA, 4–5 June 2015; pp. 806–814.
83. Bethard, S.; Savova, G.; Chen, W.T.; Derczynski, L.; Pustejovsky, J.; Verhagen, M. Semeval-2016 task 12: Clinical tempeval. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, CA, USA, 16–17 June 2016; pp. 1052–1062.
84. Bethard, S.; Savova, G.; Palmer, M.; Pustejovsky, J. SemEval-2017 Task 12: Clinical TempEval. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Vancouver, BC, Canada, 3–4 August 2017; pp. 565–572. [[CrossRef](#)]
85. Caselli, T.; Miller, B.; van Erp, M.; Vossen, P.; Palmer, M.; Hovy, E.; Mitamura, T.; Caswell, D. Events and Stories in the News. In Proceedings of the Events and Stories in the News Workshop, Vancouver, BC, Canada, 4 August 2017.
86. Naik, A.; Breittfeller, L.; Rose, C. TDDiscourse: A dataset for discourse-level temporal ordering of events. In Proceedings of the 20th Annual SIGDial Meeting on Discourse and Dialogue, Stockholm, Sweden, 11–13 September 2019; pp. 239–249.
87. Hong, Y.; Zhang, T.; O’Gorman, T.; Horowitz-Hendler, S.; Ji, H.; Palmer, M. Building a cross-document event-event relation corpus. In Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016), Berlin, Germany, 11 August 2016; pp. 1–6.
88. Bittar, A.; Amsili, P.; Denis, P.; Danlos, L. French TimeBank: An ISO-TimeML annotated reference corpus. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; pp. 130–134.
89. Costa, F.; Branco, A. TimeBankPT: A TimeML annotated corpus of Portuguese. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12), Istanbul, Turkey, 23–25 May 2012; pp. 3727–3734.
90. Jeong, Y.S.; Kim, Z.M.; Do, H.W.; Lim, C.G.; Choi, H.J. Temporal information extraction from Korean texts. In Proceedings of the Nineteenth Conference on Computational Natural Language Learning, Beijing, China, 30–31 July 2015; pp. 279–288.
91. Wonsever, D.; Rosá, A.; Malcuori, M.; Moncecchi, G.; Descoins, A. Event annotation schemes and event recognition in spanish texts. In Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, New Delhi, India, 11–17 March 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 206–218.
92. Forăscu, C.; Tufiş, D. Romanian TimeBank: An annotated parallel corpus for temporal information. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12), Istanbul, Turkey, 23–25 May 2012; pp. 3762–3766.
93. Caselli, T.; Lenzi, V.B.; Sprugnoli, R.; Pianta, E.; Prodanof, I. Annotating events, temporal expressions and relations in Italian: The It-TimeML experience for the Ita-TimeBank. In Proceedings of the 5th Linguistic Annotation Workshop, Portland, OR, USA, 23–24 June 2011; pp. 143–151.
94. Llorens, H.; Chambers, N.; UzZaman, N.; Mostafazadeh, N.; Allen, J.; Pustejovsky, J. Semeval-2015 task 5: Qa tempeval-evaluating temporal information understanding with question answering. In Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, CO, USA, 4–5 June 2015; pp. 792–800.
95. Chen, W.; Wang, X.; Wang, W.Y. A Dataset for Answering Time-Sensitive Questions. *arXiv* **2021**, arXiv:2108.06314.
96. Li, L.; Wan, J.; Zheng, J.; Wang, J. Biomedical event extraction based on GRU integrating attention mechanism. *BMC Bioinform.* **2018**, *19*, 93–100. [[CrossRef](#)] [[PubMed](#)]
97. Cinelli, L.P.; de Oliveira, J.F.; de Pinho, V.M.; Passos, W.L.; Padilla, R.; Braz, P.F.; Galves, B.; Dalvi, D.P.; Lewenfus, G.; Ferreira, J.O.; et al. Automatic event identification and extraction from daily drilling reports using an expert system and artificial intelligence. *J. Pet. Sci. Eng.* **2021**, *205*, 108939. [[CrossRef](#)]
98. Liu, J.; Huang, X. Forecasting crude oil price using event extraction. *IEEE Access* **2021**, *9*, 149067–149076. [[CrossRef](#)]
99. Event. *Oxford English Dictionary*; Oxford University Press: Oxford, UK, 2022.
100. Caufield, J.H.; Zhou, Y.; Bai, Y.; Liem, D.A.; Garlid, A.O.; Chang, K.W.; Sun, Y.; Ping, P.; Wang, W. A comprehensive typing system for information extraction from clinical narratives. *medRxiv* **2019**. [[CrossRef](#)]
101. Ebner, S.; Xia, P.; Culkin, R.; Rawlins, K.; Van Durme, B. Multi-Sentence Argument Linking. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020.
102. Guda, V.; Sanampudi, S.K. Rules based event extraction from natural language text. In Proceedings of the 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 20–21 May 2016; pp. 9–13.
103. Piskorski, J.; Tanev, H.; Atkinson, M.; Goot, E.v.d.; Zavarella, V. Online news event extraction for global crisis surveillance. In *Transactions on Computational Collective Intelligence V*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 182–212.
104. Iqbal, K.; Khan, M.Y.; Wasi, S.; Mahboob, S.; Ahmed, T. On extraction of event information from social text streams: An unpretentious nlp solution. *IJCSNS* **2019**, *19*, 1.
105. Cohen, K.B.; Verspoor, K.; Johnson, H.L.; Roeder, C.; Ogren, P.V.; Baumgartner, W.A., Jr.; White, E.; Tipney, H.; Hunter, L. High-Precision biological event extraction: Effects of system and of data. *Comput. Intell.* **2011**, *27*, 681–701. [[CrossRef](#)]

106. Kovačević, A.; Dehghan, A.; Filannino, M.; Keane, J.A.; Nenadic, G. Combining rules and machine learning for extraction of temporal expressions and events from clinical narratives. *J. Am. Med. Inform. Assoc.* **2013**, *20*, 859–866. [[CrossRef](#)]
107. Hong, J.; Davoudi, A.; Yu, S.; Mowery, D.L. Annotation and extraction of age and temporally-related events from clinical histories. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 1–15. [[CrossRef](#)]
108. Baradaran, R.; Minaei-Bidgoli, B. Event Extraction from Classical Arabic Texts. *Int. Arab J. Inf. Technol.* **2015**, *12*, 494–502.
109. Saha, S.; Majumder, A.; Hasanuzzaman, M.; Ekbal, A. Bio-molecular event extraction using Support Vector Machine. In Proceedings of the 2011 Third International Conference on Advanced Computing, Seoul, Republic of Korea, 27–29 September 2011; pp. 298–303.
110. Sinha, D.; Garain, U.; Bandyopadhyay, S. Event extraction from cancer genetics literature. In Proceedings of the 2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR), Kolkata, India, 4–7 January 2015; pp. 1–6.
111. Li, Z.; Liu, F.; Antieau, L.; Cao, Y.; Yu, H. Lancet: A high precision medication event extraction system for clinical text. *J. Am. Med. Inform. Assoc.* **2010**, *17*, 563–567. [[CrossRef](#)] [[PubMed](#)]
112. Abdulkadhar, S.; Bhasuran, B.; Natarajan, J. Multiscale Laplacian graph kernel combined with lexico-syntactic patterns for biomedical event extraction from literature. *Knowl. Inf. Syst.* **2021**, *63*, 143–173. [[CrossRef](#)]
113. Jain, P.; Bendapudi, H.; Rao, S. EEQUEST: An event extraction and query system. In Proceedings of the 9th Annual ACM India Conference, Gandhinagar, India, 21–23 October 2016; pp. 59–66.
114. Smadi, M.; Qawasmeh, O. A supervised machine learning approach for events extraction out of Arabic tweets. In Proceedings of the 2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS), Valencia, Spain, 15–18 October 2018; pp. 114–119.
115. Miwa, M.; Ananiadou, S. Adaptable, high recall, event extraction system with minimal configuration. *BMC Bioinform.* **2015**, *16*, 1–11. [[CrossRef](#)] [[PubMed](#)]
116. Han, X.; Kim, J.J.; Kwok, C.K. Active learning for ontological event extraction incorporating named entity recognition and unknown word handling. *J. Biomed. Semant.* **2016**, *7*, 1–18. [[CrossRef](#)] [[PubMed](#)]
117. Majumder, A.; Ekbal, A.; Naskar, S.K. Bio-molecular event extraction by integrating multiple event-extraction systems. *Sādhanā* **2019**, *44*, 1–7. [[CrossRef](#)]
118. Yi, S.X.; Li, C.Y. Exploring Multiple Embedded Features on Event Extraction. *J. Phys. Conf. Ser.* **2019**, *1267*, 012033. [[CrossRef](#)]
119. Yu, W.; Yi, M.; Huang, X.; Yi, X.; Yuan, Q. Make it directly: Event extraction based on tree-LSTM and bi-GRU. *IEEE Access* **2020**, *8*, 14344–14354. [[CrossRef](#)]
120. Wang, Y.; Xu, Z.; Bai, L.; Wan, Y.; Cui, L.; Zhao, Q.; Hancock, E.R.; Philip, S.Y. Cross-Supervised Joint-Event-Extraction with Heterogeneous Information Networks. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 278–285.
121. Zeng, Y.; Yang, H.; Feng, Y.; Wang, Z.; Zhao, D. A convolution BiLSTM neural network model for Chinese event extraction. In *Natural Language Understanding and Intelligent Applications*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 275–287.
122. Sahoo, S.K.; Saha, S.; Ekbal, A.; Bhattacharyya, P. A platform for event extraction in hindi. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020, pp. 2241–2250.
123. Guo, K.; Jiang, T.; Zhang, H. Knowledge graph enhanced event extraction in financial documents. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 1322–1329.
124. Wu, X.; Wang, T.; Fan, Y.; Yu, F. Chinese Event Extraction via Graph Attention Network. *Trans. Asian Low-Resour. Lang. Inf. Process.* **2022**, *21*, 1–12. [[CrossRef](#)]
125. He, X.; Yu, B.; Ren, Y. SWACG: A Hybrid Neural Network Integrating Sliding Window for Biomedical Event Trigger Extraction. *J. Imaging Sci. Technol.* **2021**, *65*, 60502. [[CrossRef](#)]
126. Ju, M.; Nguyen, N.T.; Miwa, M.; Ananiadou, S. An ensemble of neural models for nested adverse drug events and medication extraction with subwords. *J. Am. Med. Inform. Assoc.* **2020**, *27*, 22–30. [[CrossRef](#)]
127. Gao, J.; Luo, X.; Wang, H.; Wang, Z. Causal Event Extraction using Iterated Dilated Convolutions with Semantic Convolutional Filters. In Proceedings of the 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI), Washington, DC, USA, 1–3 November 2021; pp. 619–623.
128. El-allaly, E.D.; Sarrouiti, M.; En-Nahni, N.; El Alaoui, S.O. MTTLADE: A multi-task transfer learning-based method for adverse drug events extraction. *Inf. Process. Manag.* **2021**, *58*, 102473. [[CrossRef](#)]
129. Magge, A.; Tutubalina, E.; Miftahutdinov, Z.; Alimova, I.; Dirkson, A.; Verberne, S.; Weissenbacher, D.; Gonzalez-Hernandez, G. DeepADEMiner: A deep learning pharmacovigilance pipeline for extraction and normalization of adverse drug event mentions on Twitter. *J. Am. Med. Inform. Assoc.* **2021**, *28*, 2184–2192. [[CrossRef](#)] [[PubMed](#)]
130. Lybarger, K.; Ostendorf, M.; Thompson, M.; Yetisgen, M. Extracting COVID-19 diagnoses and symptoms from clinical text: A new annotated corpus and neural event extraction framework. *J. Biomed. Inform.* **2021**, *117*, 103761. [[CrossRef](#)] [[PubMed](#)]
131. Fan, B.; Fan, W.; Smith, C. Adverse drug event detection and extraction from open data: A deep learning approach. *Inf. Process. Manag.* **2020**, *57*, 102131. [[CrossRef](#)]
132. Zhang, Z.; Kong, X.; Liu, Z.; Ma, X.; Hovy, E. A two-step approach for implicit event argument detection. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 7479–7485.
133. Li, B.; Fang, G.; Yang, Y.; Wang, Q.; Ye, W.; Zhao, W.; Zhang, S. Evaluating ChatGPT’s Information Extraction Capabilities: An Assessment of Performance, Explainability, Calibration, and Faithfulness. *arXiv* **2023**, arXiv:2304.11633.

134. Wei, X.; Cui, X.; Cheng, N.; Wang, X.; Zhang, X.; Huang, S.; Xie, P.; Xu, J.; Chen, Y.; Zhang, M.; et al. Zero-shot information extraction via chatting with chatgpt. *arXiv* **2023**, arXiv:2302.10205.
135. Gao, J.; Zhao, H.; Yu, C.; Xu, R. Exploring the feasibility of chatgpt for event extraction. *arXiv* **2023**, arXiv:2303.03836.
136. Zhang, T.; Whitehead, S.; Zhang, H.; Li, H.; Ellis, J.; Huang, L.; Liu, W.; Ji, H.; Chang, S.F. Improving event extraction via multimodal integration. In Proceedings of the 25th ACM international conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 270–278.
137. Dorr, B.J.; Gaasterland, T. Exploiting aspectual features and connecting words for summarization-inspired temporal-relation extraction. *Inf. Process. Manag.* **2007**, *43*, 1681–1704. [[CrossRef](#)]
138. Yuan, C.; Xie, Q.; Ananiadou, S. Zero-shot temporal relation extraction with chatgpt. *arXiv* **2023**, arXiv:2304.05454.
139. Chan, C.; Cheng, J.; Wang, W.; Jiang, Y.; Fang, T.; Liu, X.; Song, Y. Chatgpt evaluation on sentence level relations: A focus on temporal, causal, and discourse relations. *arXiv* **2023**, arXiv:2304.14827.
140. Shi, Y.; Xiao, Y.; Quan, P.; Lei, M.; Niu, L. Document-level relation extraction via graph transformer networks and temporal convolutional networks. *Pattern Recognit. Lett.* **2021**, *149*, 150–156. [[CrossRef](#)]
141. Zhao, S.; Li, L.; Lu, H.; Zhou, A.; Qian, S. Associative attention networks for temporal relation extraction from electronic health records. *J. Biomed. Inform.* **2019**, *99*, 103309. [[CrossRef](#)] [[PubMed](#)]
142. Lin, C.; Miller, T.; Dligach, D.; Sadeque, F.; Bethard, S.; Savova, G. A BERT-based one-pass multi-task model for clinical temporal relation extraction. In Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing, Online, 9 July 2020.
143. Kanev, A.; Terekhov, V.; Chernenky, V.; Proletarsky, A. Metagraph Knowledge Base and Natural Language Processing Pipeline for Event Extraction and Time Concept Analysis. In Proceedings of the 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), St. Petersburg, Russia, 26–29 January 2021; pp. 2104–2109.
144. Ray, P.P. ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet Things-Cyber-Phys. Syst.* **2023**, *3*, 121–154. [[CrossRef](#)]
145. Labrak, Y.; Rouvier, M.; Dufour, R. A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical and biomedical tasks. *arXiv* **2023**, arXiv:2307.12114.
146. Liu, W.; Zhou, P.; Zhao, Z.; Wang, Z.; Ju, Q.; Deng, H.; Wang, P. K-bert: Enabling language representation with knowledge graph. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 2901–2908.
147. Gottschalk, S.; Demidova, E. EventKG: A multilingual event-centric temporal knowledge graph. In Proceedings of the European Semantic Web Conference, Crete, Greece, 3–7 June 2018; pp. 272–287.
148. Mo, C.; Wang, Y.; Jia, Y.; Liao, Q. Survey on temporal knowledge graph. In Proceedings of the 2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC), Shenzhen, China, 9–11 October 2021; pp. 294–300.
149. Song, Y. Construction of Event Knowledge Graph based on Semantic Analysis. *Teh. Vjesn.* **2021**, *28*, 1640–1646.
150. Leetaru, K.; Schrodt, P.A. Gdelt: Global data on events, location, and tone, 1979–2012. In Proceedings of the ISA Annual Convention, San Francisco, CA, USA, 3–6 April 2013; Volume 2, pp. 1–49.
151. Zwolski, K. Integrating crisis early warning systems: Power in the community of practice. *J. Eur. Integr.* **2016**, *38*, 393–407. [[CrossRef](#)]
152. Gottschalk, S.; Demidova, E.; Bernacchi, V.; Rogers, R. Ongoing events in Wikipedia: A cross-lingual case study. In Proceedings of the 2017 ACM on Web Science Conference, Troy, NY, USA, 25–28 June 2017; pp. 387–388.
153. Rogers, R. *Digital Methods*; MIT Press: Cambridge, MA, USA, 2013.
154. Singhal, P.; Guare, L.; Morse, C.; Lucas, A.; Byrska-Bishop, M.; Guerraty, M.A.; Kim, D.; Ritchie, M.D.; Verma, A. DETECT: Feature extraction method for disease trajectory modeling in electronic health records. *AMIA Summits Transl. Sci. Proc.* **2023**, *2023*, 487.
155. Fatemi, M.; Killian, T.W.; Subramanian, J.; Ghassemi, M. Medical dead-ends and learning to identify high-risk states and treatments. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 4856–4870.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.