Project Number: **770299**

NewsEye:

A Digital Investigator for Historical Newspapers

Research and Innovation Action
Call H2020-SC-CULT-COOP-2016-2017

D3.8: Event Detection (final)

Due date of deliverable: M45 (31 January 2022)

Actual submission date: 31 January 2022

Start date of project: 1 May 2018 **Duration:** 45 months

Partner organization name in charge of deliverable: ULR

Project co-funded by the European Commission within Horizon 2020		
Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	-
RE	Restricted to a group specified by the Consortium (including the Commission Services)	-
CO	Confidential, only for members of the Consortium (including the Commission Services)	-

Revision History

Document administrative information		
Project acronym:	NewsEye	
Project number:	770299	
Deliverable number:	D3.8	
Deliverable full title:	Event Detection (final)	
Deliverable short title:	Event Detection (final)	
Document identifier:	NewsEye-T33-D38-EventDetection-Submitted-v6.0	
Lead partner short name:	ULR	
Report version:	V6.0	
Report preparation date:	31.01.2022	
Dissemination level:	PU	
Nature:	Report	
Lead author:	Emanuela Boros (ULR)	
Co-authors:	Antoine Doucet (ULR)	
Internal reviewers:	Sarah Oberbichler (UIBK-ICH), Lidia Pivovarova (UH-CS)	
Status:		Draft
		Final
	x	Submitted

The NewsEye Consortium partner responsible for this deliverable has addressed all comments received, making changes as necessary. Changes to this document are detailed in the change log table below.

Change Log

Date	Version	Editor	Summary of changes made
22/03/2021	1.0	Emanuela Boros (ULR)	Final draft, submitted for internal review
19/04/2021	2.0	Emanuela Boros (ULR)	Final version
28/04/2021	3.0	Antoine Doucet (ULR)	Minor modifications towards submission
10/12/2021	4.0	Emanuela Boros (ULR)	Draft update, proofread and ready for internal review
10/01/2022	5.0	Emanuela Boros (ULR)	Final update, taking internal reviews into account
31/01/2022	6.0	Antoine Doucet (ULR)	Minor adjustments and submission

Executive summary

Task T3.3 is concerned with event detection (ED) in the context of large collections of digitised historical newspapers written in several languages. This public deliverable presents the work achieved in ED, the detection of novel events from news articles, with the goal to attach markup to articles (or segments of text, in the absence of article separation) that contain an event. A key challenge of ED resides in the practical issues related to the high cost of manual annotation of texts (e.g., human resources) which also implies the annotated data scarcity in a multilingual context. Since the NewsEye data consists of very large document collections of digitised historical newspapers in different languages, other important challenges that have to be overcome are the level of degradation of historical documents and the quality of the automatic text recognition (ATR) process that might hinder the performance of an event detection system. Moreover, the presence of varying spellings and variations that are found in historical newspapers needs to be approached. Thus, one aim of this task was to use language-independent models that are robust to noise, and that can, to some extent, mitigate problems caused by the low quality of the text or the appearance of historical spelling variants in multilingual digitised documents.

Contents

1 Introduction

Event extraction is an application of information extraction (IE) that implies the extraction of specific knowledge from certain incidents from texts. This task is focused on obtaining event-related information from texts, and, as commonly defined in the field of IE, it consists of two main sub-tasks. The first sub-task involves event detection (ED) that deals with the extraction of critical information regarding an event, that can be represented by a keyword, a phrase, a sentence or a span of text, which evoke that event. For example, an article can talk about a new epidemic outbreak, or about the election of a new president, where the events to be detected are represented by the name of the epidemic, or by the word 'election'. The second sub-task, mostly referred to as event argument extraction, concentrates on the extraction of event extents referring to more details about the events, such as their arguments. They often refer to the participants in the event. For example, the location of the epidemic event, the name of the president, the country of the election, are to be detected in this sub-task.

Therefore, event extraction is responding to the 5W1H questions (who did what, when, where, why, and how), questions that are capable of describing the presence of events in an article. The choice of events as a pivotal notion is motivated by the consideration that events are a natural structuring concept, at both a representation and a linguistic level, as they tie together time, space, and participants. This observation holds true especially when dealing with historical texts.

In the context of the NewsEye project, the goal of Task T3.3 was to detect the articles that contain novel events and to extract information by adding markup to the relevant documents. As suggested by the name of the task, our focus is on the first sub-task, more exactly, on event detection.

We approached this task by first distinguishing between two event definitions designated by two datasets. The DAnIEL dataset [1] consists of multilingual collected documents from different press threads in the field of health (Google News) focused on epidemic events, annotated at document level with a disease–location pair. The automatic content extraction (ACE) 2005 dataset[1] covers the most common events and event types at a more fine-grained manner of English national and international news from a variety of sources selected from broadcast news programs, newspapers, newswire reports, internet sources or transcribed audio, annotated at word level with the possibility of multiple events to be mentioned in the same document.

In this final version of the deliverable, we present several approaches to event detection, all with the ability of handling multilingual data. We chose two baseline models: one based on the DAnIEL system, which exploits the global structure of news regarding only epidemic outbreaks, and a neural-based approach that consisted in a convolutional neural network (CNN) applied to a local context around potential event triggers, independent of the type of data. We experimented on how well the models perform in perfect conditions and also, with added noise from aging documents, scanning and OCR process that can affect the quality of these event detection systems, with regard to the chosen datasets.

For our baselines, in order to tackle the lack of annotated digitised data, the annotation cost, data scarcity in regard to digitised documents, and to analyse the effect of these errors on the event detection task, we simulated the existence of the level of degradation of historical documents and the historical spellings that are found in old newspapers. We created synthetic data starting from the initial datasets in order to study the direct impact of automatic text recognition (ATR) over the performance of both approaches, in order to have a robust inspection of the challenges of the project. We conclude that both systems can be effected by the ATR errors, depending on their ability of data representation, event types, and imbalance.

Next, we present more advanced models that we developed, based on the Transformer encoder [2] and pre-trained and fine-tuned on the task language models (e.g., BERT [3]). Moreover, we also propose improvements to these models by including the influence of entities in detecting events, and by treating the event detection task as two paradigms, a sequential data classification task and question answering (QA) task.

Considering that the ACE2005 contains a complete set of finer-grained event types that can be explored in historical documents, we continue by utilising them with our more advanced developed methods. For assessing the ability of our new developed methods of handling multilingual digitised document, we evaluate them on two annotated NewsEye datasets, in two NewsEye languages, for French and for German, selected and annotated by the digital humanities (DH) team. In order to adapt our models to the NewsEye languages, we utilised pre-trained multilingual models. Due to the lack of annotated data, we adopted a zero-shot technique by training our models on the ACE2005 dataset, a dataset that contains a more detailed and fine-grained set of event types, and predicting on the datasets in the NewsEye languages. The experiments proved that, not only that our proposed models are able to detect events even though they were never seen in the specified language, but also with a high precision.

The content of this report is organised as follows. Section 3 summarises the state of the art on event extraction (EE). Section 4 introduces the two datasets that will be used for the experiments in this report. Section 5 presents in detail the main approaches for ED, and the experimental setups for both approaches are elaborated in Section 6. This setup is focused on the ED task with regard to the models

[1] https://catalog.ldc.upenn.edu/ldc2006t06

and the datasets. Section 7 presents our performance on the French and German NewsEye datasets, and we conclude the deliverable in Section 8.

2 State of the art

The area of information extraction (IE) has the task of finding these relevant data in large sets of documents and also to store them in an appropriate form. More exactly, IE is the task of automatically extracting *entities*, *relations*, and *events* from unstructured texts. The architecture of IE systems was described by the authors of [4] in the historical framework of the Message Understanding Conference [5]. Although the techniques have obviously evolved in the meantime, the main lines of processing of scanning a text for relevant information imply three levels of extraction tasks: *named entity recognition* (NER), *named entity linking* (NEL), *relation extraction* (RE), and *event extraction* (EE). NER represents the detection of target entities in text, previously presented in *D3.2: Named entity recognition and linking*, RE is the identification of binary relations between entities (e.g., individuals, or locations), NEL is the task of assigning an identity to entities (e.g., famous individuals), also previously presented in Task T3.1, and EE is a complex task that involves identifying instances of specified types of events in texts and the corresponding arguments (participants). It can benefit of the previous tasks (NER, RE).

Event extraction is useful for many practical applications, such as news summarisation and information retrieval. The research in EE has progressed through a long history that started around 1987 with the first campaign named message understanding conferences (MUC) [5] that lasted until 1998 with the help of the US government (ARPA/DARPA), followed by the automatic content extraction (ACE) program [6], and more recently by the text analysis conferences (TAC)[2]. Another known evaluation campaign was initiated in 2004 by the Informatics for Integrating Biology and the Bedside (i2b2) for the extraction of medication-related information from narrative patient records, in order to accelerate the translation of clinical findings into novel diagnostics and prognostics. In the context of event extraction, the timeline created by [7] from Digital Humanities Group at Fondazione Bruno Kessler[3], built by collecting information from websites and proceedings, summarises the history of workshops, in the lower part, and evaluation campaigns, in the upper part.

2.1 Event Extraction

Several event definitions have been proposed over the years, each showing specific strengths and weaknesses. The event detection task is challenging due to the ambiguous nature of the concept of event.

In the first campaign related to event extraction, the message understanding conference (MUC-3) in 1991, the task of EE was seen as a *template* with slots to be automatically filled with participants, time and space, and other details. The articles in the MUC dataset focused on events about terrorist attacks and violent acts perpetrated with political aims and a motive of intimidation. An article could contain multiple events, from a pre-set list of event types e.g., *bombing, attack, kidnapping* with multiple arguments, e.g., *human target*, *perpetrator*. The task of event extraction was defined as the extraction of *templates* as shown in Figure 1, where the incident is *kidnapping*, from the incident category *terrorist attack*, with different human targets (e.g., *Federico Estrada Velez*), the date (*03 April 90*) and the location (*Colombia*) of the incident.

[2] https://tac.nist.gov/about/index.html
[3] http://dhlab.fbk.eu/Timeline_events/

TST1-MUC3-0080

BOGOTA, 3 APR 90 (INRAVISION TELEVISION CADENA 1) -- [REPORT] [JORGE ALONSO SIERRA VALENCIA] [TEXT] LIBERAL SENATOR FEDERICO ESTRADA VELEZ WAS KIDNAPPED ON 3 APRIL AT THE CORNER OF 60TH AND 48TH STREETS IN WESTERN MEDELLIN, ONLY 100 METERS FROM A METROPOLITAN POLICE CAI [IMMEDIATE ATTENTION CENTER]. THE ANTIOQUIA DEPARTMENT LIBERAL PARTY LEADER HAD LEFT HIS HOUSE WITHOUT ANY BODYGUARDS ONLY MINUTES EARLIER. AS HE WAITED FOR THE TRAFFIC LIGHT TO CHANGE, THREE HEAVILY ARMED MEN FORCED HIM TO GET OUT OF HIS CAR AND GET INTO A BLUE RENAULT.

HOURS LATER, THROUGH ANONYMOUS TELEPHONE CALLS TO THE METROPOLITAN POLICE AND TO THE MEDIA, THE EXTRADITABLES CLAIMED RESPONSIBILITY FOR THE KIDNAPPING. IN THE CALLS, THEY ANNOUNCED THAT THEY WILL RELEASE THE SENATOR WITH A NEW MESSAGE FOR THE NATIONAL GOVERNMENT.

LAST WEEK, FEDERICO ESTRADA VELEZ HAD REJECTED TALKS BETWEEN THE GOVERNMENT AND THE DRUG TRAFFICKERS.

0. MESSAGE ID	TST1-MUC3-0080
1. TEMPLATE ID	1
2. DATE OF INCIDENT	03 APR 90
3. TYPE OF INCIDENT	KIDNAPPING
4. CATEGORY OF INCIDENT	TERRORIST ACT
5. PERPETRATOR: ID OF INDIV(S)	"THREE HEAVILY ARMED MEN"
6. PERPETRATOR: ID OF ORG(S)	"THE EXTRADITABLES" / "EXTRADITABLES"
7. PERPETRATOR: CONFIDENCE	CLAIMED OR ADMITTED: "THE EXTRADITABLES" / "EXTRADITABLES"
8. PHYSICAL TARGET: ID(S)	*
9. PHYSICAL TARGET: TOTAL NUM	*
10. PHYSICAL TARGET: TYPE(S)	*
11. HUMAN TARGET: ID(S)	"FEDERICO ESTRADA VELEZ" ("LIBERAL SENATOR" / "ANTIOQUIA DEPARTMENT LIBERAL PARTY LEADER" / "SENATOR" / "LIBERAL PARTY LEADER" / "PARTY LEADER")
12. HUMAN TARGET: TOTAL NUM	1
13. HUMAN TARGET: TYPE(S)	GOVERNMENT OFFICIAL / POLITICAL FIGURE: "FEDERICO ESTRADA VELEZ"
14. TARGET: FOREIGN NATION(S)	-
15. INSTRUMENT: TYPE(S)	*
16. LOCATION OF INCIDENT	COLOMBIA: MEDELLIN (CITY)
17. EFFECT ON PHYSICAL TARGET(S)	*
18. EFFECT ON HUMAN TARGET(S)	-

Figure 1: Example of MUC-3 *template* [8]

The MUC campaigns [5] lasted from 1987 through 1998 under the auspices of the US government (ARPA/DARPA). They were represented by the complexity of the task definition: the complexity of texts to be processed, the high number of slots to be filled, the need for world knowledge to fill in some of these slots.

While the MUC definition of an event consisted in the extraction of the type of event and the event participants, without making a difference between these two tasks, the automatic content extraction (ACE) competitions were rather different. Indeed, the EE task was defined as two separate sub-tasks: ED, that implies identifying instances of specified types of events in text, and event argument extraction, which is the extraction of the arguments associated to them. In the ED sub-task, each event is represented by a phrase, a sentence or a span of text, the *event trigger* (most often single verbs or phrasal verbs, but also nouns, phrasal nouns, pronouns and adverbs), which evokes that event. An example is provided in the Figure 2. After the detection and classification of the triggers, in the second sub-task, the arguments of the event must be found. Event arguments are entity mentions or temporal expressions that are involved in an event (as participants). The ACE 2005 dataset was built around specific domains, and thus, for example, 48% of the events in the training corpus belong to the *Attack* subtype [7].

```
<event ID="APW_ENG_20030520.0757-EV8" TYPE="Conflict" SUBTYPE="Attack" MODALITY="Other"
  <event_argument REFID="APW_ENG_20030520.0757-E18" ROLE="Attacker"/>
  <event_argument REFID="APW_ENG_20030520.0757-E9" ROLE="Place"/>
  <event_mention ID="APW_ENG_20030520.0757-EV8-1">
    <extent>
      <charseq START="1392" END="1477">Osama bin Laden's Al-Qaeda group possibly
launching fresh attacks in the United States</charseq>
    </extent>
    <ldc_scope>
      <charseq START="1305" END="1516">Earlier this week, Saudi and U.S. officials said they had new
intelligence pointing to Osama bin Laden's Al-Qaeda group possibly
launching fresh attacks in the United States or against American
interests overseas</charseq>
    </ldc_scope>
    <anchor>
      <charseq START="1450" END="1456">attacks</charseq>
    </anchor>
    <event_mention_argument REFID="APW_ENG_20030520.0757-E18-42" ROLE="Attacker">
      <extent>
        <charseq START="1392" END="1423">Osama bin Laden's Al-Qaeda group</charseq>
      </extent>
    </event_mention_argument>
    <event_mention_argument REFID="APW_ENG_20030520.0757-E9-44" ROLE="Place">
      <extent>
        <charseq START="1461" END="1477">the United States</charseq>
      </extent>
    </event_mention_argument>
  </event_mention>
</event>
```

Figure 2: Example of ACE 2005 event annotation

Typically, an ACE event in a text is expressed by the following components:

- *Event mention*: an occurrence of an event with a particular type. These are usually sentences or phrases that describe an event. The example in the Figure 2 is an *Attack* event mention: the text talks about an attack.
- *Event trigger*: the word that most clearly expresses the event mention. The *Attack* from the figure is revealed by the event trigger word *attacks*.
- *Event argument*: an entity mention or temporal expression (e.g., *Crime*, *Job-Title*) that serves as a participant or attribute with a specific role in an event mention. Event arguments have an entity type (PER, LOC, ORG, etc.). One could identify this task as the named entity recognition (NER). The difference between named entities and event arguments is that, generally, not all the detected entities are arguments to events and there can be unrelated entity mentions. Thus, the entities that are event arguments have roles.
- *Argument role*: the relationship between an argument and the event in which it participates. The argument roles that should be extracted in this case are: *Osama bin Laden's Al-Qaeda group* that has the role of an *Attacker* and the *Place* where the event produced is *the United States*. The *Attacker* is an argument role that are specific for the *Conflict.Attack* event type.

The *event mention* and *event trigger* are notions used in ED, and the *event argument* and *argument role* are notions used in the event arguments extraction.

Shortly after, the definition of the event has undergone minor changes, and the ERE (entities, relations, events) scheme has been developed later within the DARPA DEFT program [9] in order to simplify the ACE event type definition that made the process of annotating data very challenging. ERE and ACE share the same event types and subtypes, but the ERE annotation is simplified by collapsing tags and therefore loosening the event extent and also reducing the annotation features in order to eliminate annotator confusion and to improve coherency and consistency of the dataset.

Throughout the years, MUC, ACE and TAC initiatives have been of central importance to the IE field

since they provided a set of corpora that are available to the research community for the evaluation and comparison of IE systems and approaches.

2.2 Challenges

Despite the usefulness and prospective applicability of EE (which implies the ED sub-task) in historical and digitised documents, several issues and challenges are to be overcome until an IE system is widely adopted as an effective tool in practice.

- The *annotation cost* and *data scarcity*: there are practical issues related to the high cost of manual annotation of texts (e.g., human resources). The human effort needs to be minimised while keeping the quality of an IE system. Data annotation takes advantage of a massive human expertise, and this causes labour-intensive work for data interpretation at two levels. Firstly, an IE system may use NLP resources and tools, created using lots of annotated documents and secondly, an IE system needs a higher-level of annotation of relations or events, annotations that can be complex and extremely costly.
- The *lack of multilingual approaches*: state-of-the-art systems are limited by their language-to-language approach and are difficult to adapt to new languages. The main tendency in approaching a multilingual solution has mostly been to create several monolingual systems. The ability to handle documents written in different languages is becoming a more and more important request and is essential for NewsEye, where the integration and the analysis of historical newspapers from different sources in several languages represents a main focus. In the field of epidemiological surveillance [1, 10], for example, it is especially important to detect a new event the very first time it is mentioned, and this very first occurrence will almost always happen in the local language. Thus, the requirement of multilingualism is undoubtedly the most important challenge. It would indeed be particularly harmful to have to wait for an article in English (or in another widely used language) to signal an epidemic before being able to react. English has indeed the widest monolingual coverage, and it is understandably the first language that one is tempted to use for effective monitoring.
- The *document degradation challenge*: either from the historical degraded documents or due to the fact that most of digitised documents are indexed through their transcribed version, errors arise from automatic text recognition (ATR) errors that may hinder the access to them. Though there has been an interest in studying the effect of ATR onto other IE tasks (e.g., NER, NEL) [11, 12, 13, 14, 15, 16], to our knowledge, prior to our recent work [17, 18], there was no research done on this impact on ED.
- The *context of extraction* can be also considered an issue, since the extraction of the needed information is often approached at a local level, as in the case of the detection and extraction of entities, relations or events that are fully expressed within a single sentence. Sentence-level extraction patterns are commonly used in IE systems, but an event can benefit from the global structure of news in a newswire.

3 Event Extraction Approaches

3.1 Event Extraction in Modern Documents

In order to better generalise the systems developed for the event detection task, one can divide the prior work in:

- *pattern-based systems* [19, 20, 21, 22, 23]: these approaches first acquire a set of patterns, where the patterns consist of a predicate, an event trigger, and constraints on its local syntactic context.
- *feature-based or machine learning systems* based on engineered features: [24, 25, 26, 27, 28, 29, 30, 31, 32, 33]: these approaches rely on discriminative features to build statistical models and usually require effort to develop rich sets of features.
- *neural-based approaches*: [34, 35, 36, 37, 38]: these approaches achieve relatively high performance due to their ability of learning automatic features.

Recently, there has been a lot of interest in approaching the event detection task with external resource-based models which are either feature-based [39, 40] or neural-based [41] combined with resources as in FrameNet[4] [42] which is a linguistic corpus that defines complete semantic frames and frame-to-frame relations, or event data generation as in [43, 44, 45].

Recent approaches adopt the usage of pre-trained language models or Transformer-based methods [2], and, since BERT (Bidirectional Encoder Representations from Transformers) [3] that broke records of 11 NLP tasks (part-of-speech tagging, named entity recognition, etc.) and received a lot of attention, recent advances in event detection imply architectures based on fine-tuning this type of models [44, 45]. These methods hold the state of the art for event detection. Moreover, differently from most of the previous neural-based methods, where event detection was considered as a classification task, a new paradigm was introduced [46, 47] formulating it as a question answering (QA)/machine reading comprehension (MRC) task, where events can be extracted by responding to the 5W1H questions (who did what, when, where, why, and how).

3.2 Pattern-based Approaches

Several pattern-based (rule-based) systems have been proposed to speed up the annotation process. The pattern-based approaches first acquire a set of patterns, where the patterns consist of a predicate, an event trigger, and constraints on its local syntactic context. They also include a rich set of ad-hoc lexical features (e.g., compound words, lemma, synonyms, Part-of-Speech (POS) tags), syntactic features (e.g., grammar-level features, dependency paths) and semantic features (e.g., features from a multitude of sources, WordNet[5], gazetteers) to identify role fillers. Earlier pattern-based extraction systems were developed for the MUC conferences [19, 20, 21, 23]. For instance, the *AutoSlog* system [22] automatically created extraction patterns that could be used to construct dictionaries of important elements for a particular domain and a text, where the elements of interest were manually tagged only for the training stage. Later, the authors of [21] make the observation that patterns occurring with substantially higher frequency in relevant documents than in irrelevant documents are likely to be good extraction patterns. They propose the separation between relevant and irrelevant syntactic patterns and a re-ranking of the patterns. The system named *AutoSlog-TS* attempted to overcome the necessity of having a hand-labelled input, requiring only pre-classified texts and a set of generic syntactic patterns. The main drawback of this system is the requirement of manual inspection of the patterns, which can be costly.

Many proposed approaches targeted the minimisation of human supervision with a bootstrapping technique for event extraction. The authors of [48] proposed a bootstrapping method to extract event arguments using only a small amount of annotated data. After the manual inspection of the patterns, another effort was made for performing manual filtering of resulting irrelevant patterns.

[4] https://framenet.icsi.berkeley.edu/fndrupal/
[5] https://wordnet.princeton.edu/

The approach described in [23] included another bootstrapping approach, starting with some seed patterns, using these patterns to identify some relevant documents, using these documents to identify additional patterns, etc. The authors in [49] also proposed to sort relevant from irrelevant documents using a topic description and information retrieval engine. This approach was further refined in [26], which explored alternative pattern ranking strategies. The method in [50] used a lexical database for the English language *WordNet*-based similarity to expand an initial set of event patterns. The systems in [51, 52] are built upon a sentence classifier that distinguishes between relevant and irrelevant regions and learns domain-relevant extraction patterns using a semantic affinity measure. Later, [33] takes the example trigger terms mentioned in the guidelines as seeds, and then applies an event-independent similarity-based classifier for trigger labelling.

3.3 Feature-based Approaches

Most recent event extraction frameworks are feature-based approaches applied at the sentence-level or to a larger context (e.g., document-level). Feature-based approaches rely on discriminative features to build statistical models, and usually require effort to develop rich sets of features. We also include here the works that did not make use of large sets of features, but included knowledge at discourse-level.

The feature-based approaches rely mainly on designing large effective feature sets for statistical models, ranging from *local features* [53, 54, 55] to the higher-level structures such as cross-document, cross-sentence and cross-event information (cross-* features) e.g., *global features* [56, 31, 27, 57, 29, 28]. The discrete local features include: lexical features (e.g., unigrams/bigrams of text context, lemma, synonyms, Part-of-Speech (POS) tags, Brown clusters [58]), syntactic features (e.g., dependency paths) and semantic features (e.g., features from a set of sources, WordNet [59], gazetteers). Using NLP toolkits for extracting this type of features may lead to severe error propagation, has a cost in terms of computational efficiency, and limits the application of the models to languages for which such NLP tools are available. The cross-document and cross-sentence features are usually inferred from known instances to predict the attributes of unknown instances. As an example, given an *Attack* event, the cross-event inference can predict its type by using the related events (*Die*) co-occurring with it within the same document or same sentence. Thus, information from a larger context has been adopted in order to improve the traditional sentence-level event extraction systems. The probabilistic soft logic (PSL) based approach described in [60] employs both latent local and global information for event detection.

3.4 Neural-based Approaches

The first approaches were based on convolutional and recurrent neural networks (CNNs and RNNs). Approaches presented in [35] and [34] deal with the event detection problem with a model based on CNNs. The CNN models in [61] improve the previous models [35] for event detection by taking into account the possibility to have non-consecutive n-grams as basic features instead of continuous n-grams. Both models use word embeddings for representing windows of text that are trained like the other parameters of the neural network.

The system proposed by [62] extracts event instances from health records with bidirectional recurrent neural networks (Bi-RNNs) while [36] introduces a joint framework with the same type of neural networks for predicting at the same time event triggers and their arguments. This last work is benefiting from the advantages of the two models, as well as addressing issues inherent in the existing approaches. The authors also systematically investigate the usage of memory vectors and matrices to store the prediction information during the course of labelling sentences features. Additionally, the model presented

in [36] is augmented with discrete local features inherited from [32]. The authors of [63] advocate a graph convolution network (GCN) based on dependency trees for exploiting syntactic dependency relations to perform event detection with a pooling method that relies on entity mentions to aggregate the convolution vectors.

The papers [64] and [65] explore another extension of RNNs by integrating a larger context through a document representation, while [66] exploits a generative adversarial network for discarding spurious detections. The problem of ambiguous indicators of particular types of events (the same word can express completely different events, *fired* can correspond to an *Attack* type of event, or it can express the dismissal of an employee from a job) is tackled in [67] by the usage of RNNs and cross-lingual attention (a multilingual *attention mechanism*) to model the confidence of the features provided by other languages.

Further, some researchers have proposed other hybrid neural network models, which combine different neural networks to make use of each other's abilities. A hybrid neural network (a CNN and an RNN) is developed in [37] in order to capture both sequence and chunk information from specific contexts, and use them to train an event detector for multiple languages without any handcrafted features.

3.4.1 Transformer-based Approaches

The current state-of-the-art systems for event extraction involve Transformer-based network [2] models to improve event extraction. Recently, different approaches based on the Transformer architecture have been proposed. Transformer-based generative adversarial networks (GANs) have been applied in event detection [44, 45]. Besides, reinforcement learning (RL) is used in [44] for creating an end-to-end entity and event extraction framework. The approach attempted by [43] is based on the BERT model, with an automatic generation of labeled data by editing prototypes and filtering out the labeled samples through argument replacement by ranking their quality. A similar framework was proposed by [68] where the informative features are encoded by BERT and a CNN, which would suggest a growing interest not only in language model-based approaches but also in adversarial models. The model proposed by [69] is a BERT-based architecture that models text spans and is able to capture within-sentence and cross-sentence context. Simultaneously, an integration of a distillation technique to enhance the adversarial prediction was explored in [70].

3.4.2 External Resource-based Approaches

Recent models include also additional informative features provided by the presence of entities. Most current state-of-the-art systems perform event detection individually [34, 35, 64], where the entities are either ignored or considered helpful in joint models. Some works made use of entities in different manners. Higher results can be obtained with gold-standard entity types [35], by concatenating randomly initialized embeddings for the entity types. A graph neural network (GNN) based on dependency trees [63] has also been proposed to perform event detection with a pooling method that relies on entity mentions aggregating the convolution vectors. Arguments provided significant clues to this task in the supervised attention mechanism proposed to exploit argument information explicitly for event detection [41], while also using events from FrameNet. Although some joint learning-based methods have been proposed, which tackled event detection and argument extraction simultaneously, these approaches

usually only make significant improvements on the argument extraction, but insignificant to event detection.

These methods usually combine the loss functions of these two tasks and are jointly trained under the supervision of annotated triggers and arguments. Event triggers and their arguments are predicted at the same time in a joint framework [36] with bidirectional recurrent neural networks (Bi-RNNs) and a convolutional neural network (CNN) and systematically investigate the usage of memory vectors/matrices to store the prediction information during the course of labeling sentence features. The architecture adopted in [71] was to jointly extract multiple event triggers and event arguments by introducing syntactic shortcut arcs derived from the dependency parsing trees to enhance the information flow in an attention-based graph convolution network (GCN) model. The gold-standard entity types are embedded as features for trigger and argument prediction. The argument information was also exploited in [41] explicitly for event detection by experimenting with different strategies for adding supervised attention mechanisms. The authors exploit the annotated entity information by concatenating the token embeddings with randomly initialized entity type embeddings.

In the context of event detection, some works made use of gold-standard entities in different manners. Higher results can be obtained with gold-standard entity types [35], by concatenating randomly initialized embeddings for the entity types. A graph neural network (GNN) based on dependency trees [63] has also been proposed to perform event detection with a pooling method that relies on entity mentions aggregation. Arguments provided significant clues to this task in the supervised attention mechanism proposed to exploit argument information explicitly for ED proposed by [41]. Other methods that took advantage of argument information were joint-based approaches. The architecture adopted by [71] was to jointly extract multiple event triggers and event arguments by introducing syntactic shortcut arcs derived from the dependency parsing trees.

Neural-based approaches achieve relatively high performance due to their ability of learning automatic features. However, as we mentioned before, data scarcity in ED limits their further performance. An external resource-based model tackles data scarcity problems by exploiting additional information. The authors of [33] take the example trigger terms mentioned in the guidelines as seeds, and then applies an event-independent similarity-based classifier for trigger labelling. Thus, a great amount of effort has been put in to overcome the manual annotation of data.

The approach proposed in [39] uses a probabilistic soft logic (PSL) based approach and a vanilla neural network by also leveraging the annotated corpus of the external resource FrameNet to alleviate data sparseness problem of ED based on the observation that frames in FrameNet are analogous to events. The authors of [72] also consider that arguments provide significant clues to this task, and adopt a supervised attention mechanism to exploit argument information explicitly for event detection, while also using events from FrameNet, as extra training data.

The model described in [40] also leverages FrameNet by tackling the challenge of the annotation cost and data scarcity by considering that ACE 2005 dataset defines very limited and specific event schemes and they redefine them based on FrameNet by expressing event information with frame and building a hierarchy of event schemas that are more fine-grained and have much wider coverage than ACE. However, their approach might be difficult to be adapted to other languages since FrameNet has a low coverage of other languages, not to mention under-represented languages (low-resource languages).

Recently, different approaches that include external resources have been proposed. For example,

generative adversarial networks (GANs) framework has been applied in event extraction [44, 45]. In addition, [44] used reinforcement learning (RL) for creating an end-to-end entity and event extraction framework. [43] attempts an approach based on BERT pre-trained model [3] (Pre-trained Language Model based Event Extractor (PLMEE)) with automatic generation of labelled data by editing prototypes and filtering out the labelled samples through argument replacement by ranking their quality.

3.4.3 Other Paradigms

Moreover, other paradigms have been proposed for tackling the event detection task, more exactly, treating this task as a question answering (QA) task. Since the annotation at sentence-level is costly, requires lots of expertise, and it needs to be re-done whenever we update the event ontology, extractive QA systems have been proposed to approach the event extraction task. Extractive QA is a popular task for natural language processing (NLP) research, where models must extract a short snippet from a document in order to answer a natural language question. Thus, by formulating it as a question answering (QA)/machine reading comprehension (MRC) task, events can be extracted by responding to the 5W1H questions (who did what, when, where, why, and how) [73].

While QA for event detection is roughly under-researched, Transformer-based models have led to striking gains in performance on MRC tasks recently, as measured on the SQuAD v1.1[6] [74] and SQuAD v2.0[7] [75] leaderboards.

A recent work proposed by [46] introduced this new paradigm for event extraction by formulating it as a QA task, which extracts the event triggers and arguments in an end-to-end manner. For detecting the event, they considered an approach based on BERT that is usually applied to sequential data. The task of ED is a classification-based method where the authors designed simple fixed templates as in *what is the trigger*, *trigger*, *action*, *verb*, without specifying the event type. For example, if they chose *verb* template, the input sequence would be: [CLS] *verb* [SEP] sentence [SEP]. Next, they use a sequential fine-tuned BERT for detecting event trigger candidates.

Another recent paper [47] also approaches the event extraction task as a question answering task, similar to the [46] method. The task remains classification-based (instead of the span-based QA method) for trigger extraction, jointly encode [EVENT] with the sentence to compute an encoded representation, as in the approach proposed by [46] where the special token was *verb* or *trigger*.

3.5 Event Extraction in Historical Documents

Ryan Benjamin Shaw [76] argues that *"a historian never develops this understanding 'from scratch' or 'discovers' it in the archives. Instead, he produces it by transforming inherited ideas, which may be concepts taken for granted in his culture or concepts developed by his peers and predecessors."* Following this statement, this process can be viewed as an area where the identification and classification of events can contribute to the construction of more nuanced knowledge bases that could enable further data exploration and help to shape the humanities and historians' research [77].

[6] SQuAD v1.1 consists of reference passages from Wikipedia with answers and questions constructed by annotators after viewing the passage

[7] SQuADv2.0 augmented the SQuAD v1.1 collection with additional questions that did not have answers in the referenced passage.

For example, a project proposed in 2004 involved the enhancement of materials drawn from the Franklin D. Roosevelt Library and Digital Archives and undertook the encoding, annotation, and multi-modal linkage of a portion of the collection [78]. Moreover, the authors proposed an enhancement of a Web-based interface that enables data exploitation for providing a deeper search and access methods for historians of the World War II. The documents were scanned, hand-validated, and enriched with various entities (such as person names, dates, locations, job titles), part-of-speech, and chunking information. Since for historical research the identification of a range of events is essential, the paper presents a method based on resources like FrameNet[8]. Considering that they worked in a narrow domain, primarily in the Memoranda of Conversation, the focus was only on the identification of communicative events reported in the documents. Therefore, the method implied the extraction of verbs associated with any of the FrameNet "Communication" frame and frame hierarchy. Finally, a communicative event utilised a scheme that assigned the role of communicator to a tagged person or pronoun preceding the verb, and assumes the event comprises the remainder of the sentence.

This simple method for extracting specific targeted event types continued with a computational analysis of Italian war bulletins in War World I and II [79]. This was considered a novel work since WWII Italian war bulletins had never been digitised before. Moreover, other challenges intervened as the type of language (Italian of the first half of the 20th century) and domain (military) required an intense effort of adaptation of existing NLP tools. Bulletins were automatically annotated with different types of information, such as simple and multi-word terms, named entities, events, participants, time, and georeferenced locations. In this work, instances of major event types (e.g., bombing, sinking, battles) were established before applying the FrameNet-based method [78]. The annotated texts and extracted information were also explored with a dedicated Web interface.

Another historical event extraction module was proposed to be used for museum collections [80], allowing users to search for exhibits related to particular historical events or actors within time periods and geographic areas, extracted from Dutch historical archives. The authors focused on historical event extraction from textual data about the Srebrenica Massacre, which was a recent event (July 1995) with a big impact on the public opinions [81]. They defined the event as a historical event model which consists of four slots: a location slot, time, participant, and an action slot.

Since the analysis of the past can help to understand the present and future events, research in forecasting was also proposed. One particular area of research for predictive models using open source text has been the incorporation of events involving actors of political interest. Forecasting political instability has been a central task in computational social science for decades. Effective prediction of global and local events is essential to counter-terrorist planning: more accurate prediction will enable decision-makers to allocate limited resources in a manner most likely to prove effective [82]. These events can cover a range of interactions that span the spectrum from cooperation (e.g. the United States promising aid to Burma) to conflict (e.g. al-Qaeda representatives blowing up an oil pipeline in Yemen).

4 Datasets

Datasets for historical events are currently unavailable, and thus, in this section, we present the two datasets thate we chose in order to approach the ED task:

- DAnIEL Dataset [1] destined for multilingual epidemic surveillance and which contains articles on

[8] https://framenet.icsi.berkeley.edu/fndrupal/

different press threads in the field of *health* (Google News) focused on epidemic events from different collected documents in different languages, with events simply defined as disease-location-number of victims triples.

- ACE 2005 (automatic content extraction) dataset that covers a larger set of predefined events, ACE, which contains datasets in multiple languages for the 2005 ACE evaluation[9], with 8 events types, and 33 subtypes covering the most common events of national and international news (from a variety of sources selected from broadcast news programs, newspapers, newswire reports, internet sources and from transcribed audio),

4.1 DAnIEL Dataset

The corpus was built specifically for this system [1, 10, 83], containing articles from six different languages (English, French, Greek, Russian, Chinese, and Polish). It contains articles on different press threads in the field of *health* (Google News) focused on epidemic events. These documents have lengths that vary substantially, ranging from a short dispatch with one paragraph to a long article with a more detailed structure. Annotators, native speakers of the aforementioned languages, decide whether an article is relevant (speaks about an event) or not and then provide the disease name and location of the event. [1] defines an event as at least a disease-location pair, and in rarer cases as a disease-location-number of victims triple.

4.1.1 DAnIEL Event Definition

The DAnIEL event is defined at document-level, meaning that a document is represented by a single event and annotated with a (disease, location) pair. An example is presented in Figure 3. Thus, in the context of DAnIEL, the task of ED is defined as the identification of articles that contain an event and the extraction of the disease name and location, i.e. the words or compound words that most evokes best that event. Since the events are related to epidemic outbreaks, there is no pre-set list of types and subtypes of event, and thus the task of ED is simplified to detecting the presence of a single event related to epidemics.


```
"15962": {
    "annotations": [
        [
            "listeria",
            "USA",
            "unknown"
        ]
    ],
    "comment": "",
    "date_collecte": "2012-01-12",
    "language": "en",
    "document_path": "doc_en/20120112_www.businessweek.com_2a21025f6f4dc13c9eb8ebf3d
    "url": "http://www.businessweek.com/news/2012-01-10/listeria-cantaloupe-outbreak
},
```


Figure 3: Example of an event annotated in DAnIEL dataset.

[9] https://catalog.ldc.upenn.edu/ldc2006t06

Total documents	Polish	Chinese	Russian	Greek	French	English
4,822 (489)	352 (30)	446 (16)	426 (41)	390 (26)	2,733 (340)	475 (31)

Table 1: Summary of the DAnIEL dataset. The number of documents annotated with events is reported in brackets.

Commonly for ED, the dataset is characterised by imbalance. In this case, only around 10% of these documents are relevant to epidemic events, which is very sparse. The number of documents in each language is rather balanced, except for French, having about five times more documents compared to the rest of the languages. More statistics on the corpus can be found in Table 1.

4.1.2 DAnIEL Annotation Style

The DAnIEL dataset is annotated at document-level, which differentiates it from other datasets used in research for the ED task. A document is either reporting an event (disease-place pair, and sometimes the number of victims) or not. In order for us to be able to test the different models that we proposed, we transformed this annotation to sentence-level. The annotations provided by DAnIEL at document-level are looked-up in the appropriate file and the found offsets are attached to them. For example, the article below has the following annotations, at document level: **malaria**, **worldwide**, and **655000**.

GENEVA: Malaria caused the death of an estimated 655,000 people last year, with 86 percent of victims children aged under five, World Health Organisation figures showed on Tuesday. The figure marked a five percent drop in deaths from 2009. Africa accounted for 91 percent of deaths and 81 percent of the 216 million cases worldwide in 2010. In its annual World Malaria Report for 2011, the WHO hailed as a "major achievement" a 26 percent fall in mortality rates since 2000 despite being well short of its 50 percent target. The UN health agency aims to eradicate malaria deaths altogether by the end of 2015 and reduce the number of cases by 75 percent to 2000 levels.

In this case, in the first sentence, *GENEVA: **Malaria** caused the death of an estimated 655,000 people [. . .]*, we are able to annotate **Malaria** at positions relative to the entire article 8 – 14. The process is automatic and continues in the same manner for the other annotations. In the case where one annotation is not found in the article (e.g., **655,000** is not recognised) it is disconsidered with the risk of penalty in evaluation. From a total of 1,268 (disease names, place names, and number of patients), 1,084 were identified in the DAnIEL dataset.

First, we consider the lemma of an annotated disease name that will further be looked-up in the text. If any disease name or location is found multiple times in the text, we annotate all the present instances. Sometimes, the exact surface form of a disease name cannot be found in the text, as it is the case for Russian, Greek, and Polish articles (morphologically rich languages), we considered the annotation of the grammatical cases of nouns. For example, in Russian, "Простуда" ("prostuda") means "cold", and since this disease name cannot be found in the text article, we used the instrumental case in Russian that can generally be distinguished by the "-ом" ("-om") suffix for most masculine and neuter nouns, the

"-ою/"-ой" ("-oju"/"-oj") suffix for most feminine nouns. The instrumental case for singular "простудой" was annotated in the article text.

In the case of locations, there were 57% of cases where the location could not be found in the text, mainly due to the coarse-grained type of manual annotation at the country-level. For the annotation of the locations at a finer-grained level, we considered the presence of cities or regions in the text. For example, if the document was previously annotated with "France", and "Corsica" is mentioned in the text, we changed the final annotation to "Corsica".

In order to resolve the issues produced by automatically changing the document-level annotation level to sentence-level (entities missed by the automatic process), we annotate the dataset using Doccano[10] annotation tool, a collaborative annotation tool that provides annotation features for various tasks, among them sequence labeling task. We set up six annotation projects, for the languages under consideration (French, English, Polish, Greek Russian, and Chinese) and we defined the labels and annotation guidelines. Three annotators were recruited for each language, who are native speakers of their respective languages. The annotation process entailed tasking the annotators with reading through the news text, identifying and marking the spans for the key entities that describe the occurrence of an epidemic event, that is, the disease name and the location where the disease outbreak is reported.

4.2 ACE 2005 Dataset

We used for our experiments, as most EE systems, the annotated ACE 2005 corpus provided by the ACE evaluation[11]. ACE events are restricted to a range of types, each with a set of subtypes. Thus, only the events of an appropriate type are annotated in a document. The ACE dataset contains datasets in multiple languages (Chinese, Arabic, and English) with various types annotated for entities, relations, and events, from various information sources (e.g., broadcast conversations, broadcast news and telephone conversations).

The data were created by Linguistic Data Consortium (LDC) with support from the ACE Program. The proposed tasks by ACE are more challenging than their MUC forerunners. In particular, the increased complexity resulted from the inclusion of various information sources and the introduction of more fine-grained entity types (e.g., facilities, geopolitical entities, etc.). In the context of this project, we use only the English ACE 2005 corpus that is composed of 599 articles. For the comparison of both models proposed, this dataset cannot be tested with the DAnIEL system, since it is designed only for epidemic related data.

total documents	NW	BN	BC	WL	UN	CTS
599 (553)	106 (104)	226 (211)	60 (60)	119 (93)	49 (47)	39 (38)

Table 2: English ACE 2005 corpus summary, newswire (NW), broadcast conversation (BC), broadcast news (BN), telephone speech (CTS), Usenet newsgroups (UN), and weblogs (WL). The number of documents annotated with one or multiple events is reported in brackets.

[10] https://github.com/doccano/doccano
[11] https://catalog.ldc.upenn.edu/ldc2006t06

The corpus has 8 types of events, with 33 subtypes. For a more detailed presentation of the event types and subtypes, we refer the readers to the ACE 2005 Guidelines[12]. These are the types of events:

- *Business*: Start-Org, Merge-Org, Declare-Bankruptcy, End-Org
- *Conflict*: Attack, Demonstrate
- *Contact*: Meet, Phone-Write
- *Life*: Be-Born, Marry, Divorce, Injure, Die
- *Movement*: Transport
- *Justice*: Arrest-Jail, Release-Parole, Trial-Hearing, Charge-Indict, Sue, Convict, Sentence, Fine, Execute, Extradite, Acquit, Appeal, Pardon
- *Transaction*: Transfer-Ownership, Transfer-Money
- *Personnel*: Start-Position, End-Position, Nominate, Elect

4.2.1 ACE 2005 Event Definition

An ACE event is represented by an *event mention* (a text contains an event of a specific type and subtype), *event trigger* (the word that expresses the event mention), *event argument* (a participant in the event of a specific type), *argument role* (the role that the entity has in the event).

Since the EE task in the context of ACE 2005 has two sub-tasks, the ED represents the detection of the texts that contain an event of a specific type and the extraction of the event trigger from the text that expresses that type of event, and the event argument extraction, that is the detection of entities and their role in the event.

4.2.2 ACE 2005 Annotation Style

Every document is characterised by multiple events, or no events at all. If we consider, for instance, this example from ACE 2005 dataset.

*There was the free press in Qatar, Al Jazeera, but its offices in Kabul and Baghdad were **bombed** by Americans.*, an event detection system should output:

- *event mention*: this sentence contains an event of type **Conflict** and subtype **Attack**
- *event trigger*: this event of type **Conflict** and subtype **Attack** is triggered by the word **bombed**

An event argument extraction system should output:

- the *event arguments*: *Kabul* and *Baghdad*, which are entities of type **location**, and *Americans* which are considered an entity of type **person**;
- the *event argument roles*: *Kabul* and *Baghdad* are **Place**s and *Americans* have the **Attacker** role.

As a reminder, Task T3.3 of NewsEye is concerned with event detection, and is not addressing event argument extraction.

[12] https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf

5 Approaches

We propose two extensions of BERT-based IE models. For comparison, we also run experiments with two previous state-of-the art model. We now briefly present these models.

5.1 DAnIEL System

DANIEL [1] stands for Data Analysis for Information Extraction in any Language. The approach is at discourse-level, as opposed to the commonly used analysis at sentence-level, by exploiting the global structure of news as defined by the authors of [84]. Entries in the system are news texts, title and body of text, the name of the source when available, and other metadata (e.g., date of article). As the name implies, the system has the capability to work in a multilingual setting due to the fact that it is not a word-based algorithm, which are highly language-specific, but rather a character-based one that centers around repetition and position [1]. By avoiding grammar analysis and the usage of other NLP toolkits (e.g., part-of-speech tagger, dependency parser) and by focusing on the general structure of journalistic writing style [85, 84], the system is able to detect crucial information in salient zones that are peculiar to this genre of writing: the properties of the journalistic genre, the style universals, form the basis of the analysis.

Due to the fact that the DAnIEL does not rely on any language-specific grammar analysis, and considers text as sequences of strings instead of words, DAnIEL can quickly operate on any foreign language and extract crucial information early on and improve the decision-making process. This is pivotal in epidemic surveillance since timeliness is key, and more than often, initial medical reports are in the vernacular language where patient zero appears [1].

In the approach presented in [1], the document is the main unit, and it has language-independent organisational properties. The assumption is that the document-detectable features at a document granularity offer high robustness at the multilingual scale. The author suggests using the text as a minimal unit of analysis, beyond its relation to the genre from which it came. The press article is thus of this type, which has precise rules: the structure of the press article and the vocabulary used are established and there are well-defined communication aims known to the source as well as the target of the documents. These rules, at a higher level than the grammatical rules, are very similar in different languages, and from the knowledge of these rules, remarkable positions are defined which are independent of languages.

To exploit the positions, the author of [1] got inspired by the work on gender invariants carried out by [86, 84]. In the news genre, the different positions in the text are defined here as follows:

- beginning of text: ideally composed of the title of the article;
- beginning of body: containing the first two paragraphs;
- end of body (foot): comprising the last two paragraphs;
- rest of body: made up of the rest of the textual elements (e.g., paragraphs).

For example, [1] demonstrates the fixed structure of a news article with this example:

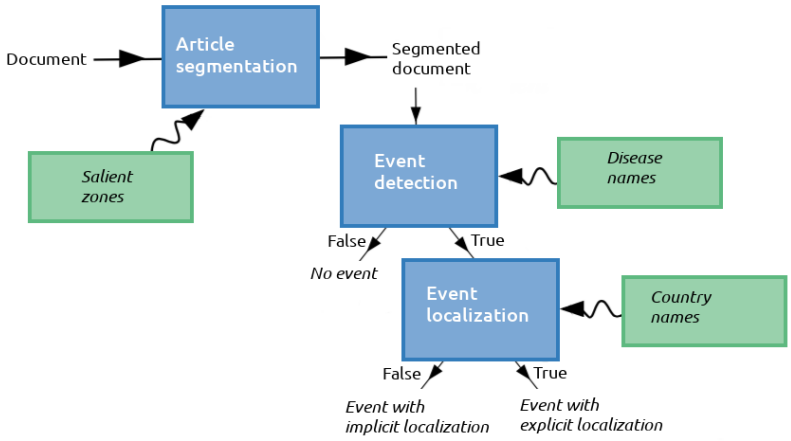

Figure 5: Event detection pipeline in DAnIEL.

Title: One was an arrogant bully. The other was a nervous wreck. So what is the truth about the war of Van Gogh's ear?
Paragraph 1/38: The iconic story about Dutch-born painter Vincent Van Gogh cutting off his own ear and presenting it as a gift to his favourite prostitute may not be true after all. **Paragraph 2/38**: Or so say some German art historians, who now claim the famous ear was cut off in a fight with rival artist Paul Gauguin.
...
Paragraph 29/38: Gauguin took himself off to Tahiti where he entertained under-age mistresses, consumed vast quantities of absinthe and morphine and died of syphilis in 1903.
...
Paragraph 37/38: Even this did not put an end to his torture. Van Gogh staggered back to the inn where he was lodging and lingered for two days before dying.
Paragraph 38/38: His poignant last words, according to Theo, the distraught brother who had rushed to his side, were: "The sadness will go on for ever."

Figure 4: Representation of the occurrences of different terms in an example from [1] in English. The name of the disease, in red that led to the classification error, is reduced. The names of the two painters in question are in blue. The constituents of the event mainly described in the article appear in orange.

In Figure 4, one can see that important pieces of information are repeated at easily identifiable positions in the text. These elements are usually found in at least two of these positions. We can see that the terms Gauguin and Van Gogh have a rich distribution. The same applies to the terms relating to Van Gogh's cut ear. Position and repetition therefore make it possible here to prioritise information without resorting to local analysis.

DAnIEL uses a minimal knowledge base, its central processing chain includes four phases, as shown in Figure 5:

- *Article segmentation*: The system first divides the document into stylistic segments: title, header, body and footer. The purpose is to identify salient zones where important information is usually repeated.
- *Pattern extraction*: For detecting events, the system will look for repeated substrings at the salient zones aforementioned and determine whether they are maximal or not. A maximal substring is a string that cannot be extended to either its left nor right side [87].

- *Filtering of these patterns*: Substrings that satisfy this condition will be matched to a list of disease/location names that was constructed by crawling from Wikipedia. The reason for using Wikipedia to build the knowledge base is that it is convenient to add lexicons from new languages without the assistance of a native speaker, since information on Wikipedia can be easily crawled from one language to another.
- *Detection of disease − location pairs* (in some cases, the number of victims also): The end result of processing a document with DAnIEL is one or more events that are described by pairs of disease-location.

5.2 Neural-based Models

5.2.1 Convolutional Neural Network-based Model

Due to their succes in detecting events, we chose a convolutional neural network (CNN) based model, inspired from [88, 35, 38] where the ED task is modelled as a word classification task.

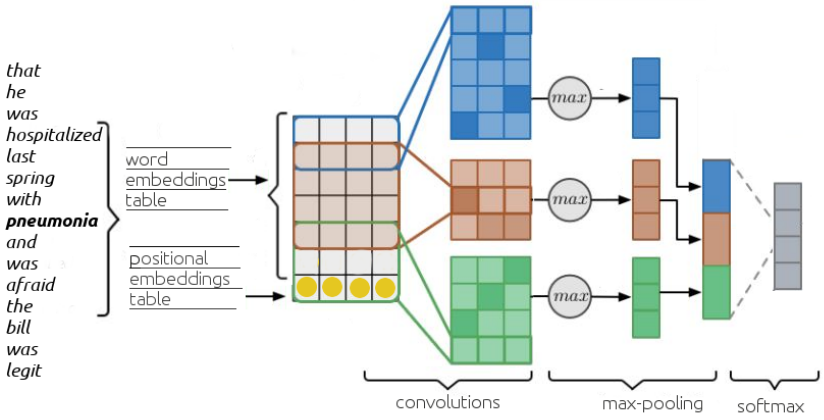

Figure 6: CNN model for ED, where *pneumonia* is the current event candidate in a context window of $2 \times n + 1$ words, where $n = 7$. Figure from [38].

Considering a sentence, we want to predict, for each word of the sentence, if the current token is a trigger of a specific type of event. The current token/word $x^{(i)}$ is surrounded by a context that constitutes the main input for the CNN. The maximum size of a sentence is established on the training data. In order to consider a limited sized context, longer sentences are trimmed, and shorter ones are padded with a special token. Let $x = [x^{(0)}, x^{(1)}, ..., x^{(N)}]$ be a sentence with words from 0 to N. Given a document, we first generate a set of event candidates $\mathcal{T}$.

For each event candidate $x^{(i)} \in \mathcal{T}$, we associate it a context window. We consider $2 \times n+1$ the size of the context window, thus a trigger candidate $x^{(0)}$ is represented as $x = [x^{(-n)}, x^{(-n+1)}, \ldots, x^{(0)}, \ldots, x^{(n-1)}, x^{(n)}]$. Each context token $x^{(i)}$ has as features the word itself and the relative position of the token to the trigger candidate $x^{(0)}$. In this case, the distance 0 will be attributed to the trigger candidate $x^{(0)}$ and $-n$, $+n$ to the marginal tokens of the window, all the other relative distances in between $-n$ and $+n$ belong to the tokens in between.

The position of an event trigger can be an informative signal for this prediction task. Each core feature

is embedded and represented in a d-dimensional space. Each feature (word, distance) is mapped to a vector retrieved from the following embedding tables:

- Word embedding table: initialised randomly or by pre-trained word embeddings (in our case, as it will be presented in the experiments section, we used the *Word2vec* for Google News [89]);
- Positional embedding table: to embed the relative distance i of the token $x^{(i)}$ to the current token $x^{(0)}$. The table is initialised randomly, and these distance embedding vectors are then trained as regular hyperparameters of the network [90, 91, 92, 35, 38].

The hyperparameters used for the CNN model for event detection are as follows. The window sizes used in the experiments are in the set $\{1, 2, 3\}$ to generate feature maps, and 300 feature maps are used for each window size in this set. After each convolutional layer, a $ReLu$ nonlinear layer is applied with orthogonal weights initialisation. The window size for triggers is also set to 31 and the dimensionality of the position embeddings is 50 [35]. The size of the batch is set to 256 and we employed also the pre-trained word embeddings *Word2vec* for Google News [89]. We would also stress the fact that the batch size affects the Adam optimizer [93], and thus our choice of 256, which performed the best on the validation set. Also, deep learning models are stochastic and use randomness (e.g., random initial weights, random shuffling) while being trained on a dataset and, because of this, a common practice is to run the algorithms several times and to report a measure of variability. Thus, we report the precision, recall and F1 in terms of means and standard deviations.

5.3 Transformer-based Models

This type of language models are neural-based architectures designed to pre-train deep bidirectional representations from unlabeled text by jointly learning both left and right context in all layers. As a result, these language models can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks, such as question answering, named entity recognition, machine translation, without substantial task-specific architecture modifications. These models are composed of a stack of Transformer layers. A Transformer block (or encoder) [2], as shown in Figure 7, is a deep learning architecture based on multi-head attention mechanisms with sinusoidal position embeddings. It is composed of a stack of identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. A residual connection is around each of the two sub-layers, followed by layer normalization. All sub-layers in the model, as well as the embedding layers, produce outputs of dimension 512. In our implementation, we used learned absolute positional embeddings [94] instead, as it is a common practice[13]. [2] found that the two versions produced nearly identical results.

Since the proposal of the first such model, BERT (Bidirectional Encoder Representations from Transformers) [3], an architecture that broke records of most NLP tasks, Transformer-based models received a lot of attention. BERT was the first fine-tuning based representation model that achieved state-of-the-art performance on a large suite of sentence-level (sentiment analysis, etc.) and token-level tasks (named entity recognition, etc.), outperforming many task-specific architectures.

The following models in this deliverable, are based on, first, fine-tuning such architectures, and second, on bringing several improvements and paradigms regarding the complexity of the models. Fine-tuning is straightforward since the self-attention mechanism in the Transformer allows BERT to model many

[13] https://huggingface.co/

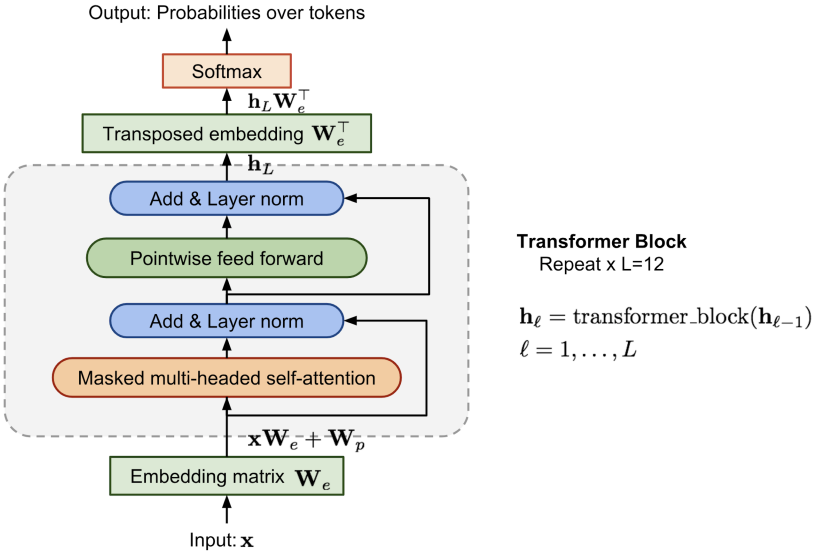

Figure 7: A Transformer block [2].

downstream tasks, whether they involve single text or text pairs by swapping out the appropriate inputs and outputs. In our case, the approaches for event detection involve token-level

5.3.1 Transformer-based Classification Model

First, our model extends the BERT [3] model applied to sequential data, as shown in Figure 8. We modify BERT by adding a conditional random fields (CRF) layer instead of the dense one, which is commonly used in other works on sequential labeling [95, 96] to ensure output consistency.

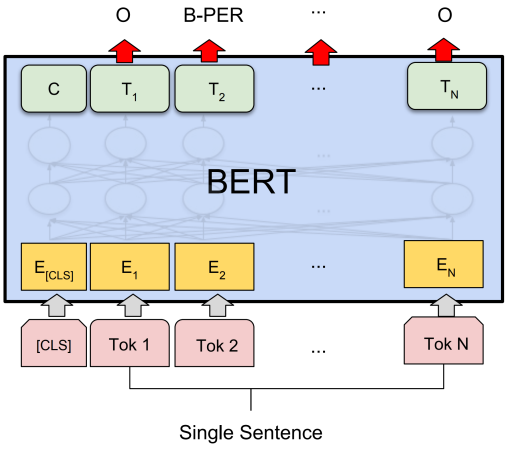

Figure 8: The BERT-based model and a CRF top layer.

5.3.2 Transformer-based Classification Model with Named Entities

Our second architecture implies the base model previously presented, but enriched with entities. We implemented the BERT-based model with *EntityMarkers*[14] We adapt the method presented in [97] applied for relation classification, to perform event detection. Next, the *EntityMarkers* model [97] consists in augmenting the input data with a series of special tokens. Thus, if we consider a sentence $x = [x_0, x_1, \ldots, x_n]$ with n tokens, we augment x with two reserved word pieces to mark the beginning and the end of each event argument mention in the sentence.

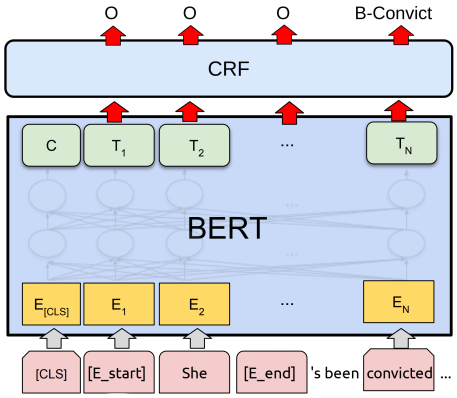

Figure 9: The BERT-based model with *Entity Position Markers* and a CRF top layer.

In the ACE 2005 dataset, an event argument is defined as an entity mention, a temporal expression or a value (e.g., *Crime*, *Sentence*, *Job-Title*) that is involved in an event (as participants or attributes with a specific role in an event mention). An event argument has an entity type and a role. For example, in a *Conflict.Attack* event type, one event argument can be an *Attacker* with three possible types: PER, ORG, GPE (Person, Organization, Geo-political Entity). Thus, we introduce three types of markers: (1) *Entity Position Markers*, e.g., $[E_{start}]$ and $[E_{end}]$ where E represents an entity of any type, (2) *Entity Type Markers*, e.g., PER_{start} and PER_{end} where PER represents an entity of type Person, and (3) we also test that, in the case of the event argument roles are known beforehand, the *Argument Role Markers*, e.g., $[Defendant_{start}]$, $[Defendant_{end}]$ where Defendant is an event argument role. We modify x to give:

$x = [x_0, x_1, \ldots, [MARKER_{start}]x_i \ldots x_{j-1}[MARKER_{end}], \ldots, x_n]$ and we feed this token sequence into BERT instead of x. We also update the entity indices $E = (i+1, j+1)$ to account for the inserted tokens, as shown in Figure 9 for the model with *Entity Position Markers*.

As an example, in the sentence "***She**'s been convicted of **obstruction of justice**.*", where *She* has the argument role of a **Defendant** and *obstruction of justice* is an argument of type **Crime**, the sentence is augmented as follows:

(1) $[E_{start}]$ ***She*** $[E_{end}]$*'s been convicted of* $[E_{start}]$ ***obstruction of justice***$[E_{end}]$.

(2) $[PER_{start}]$ ***She*** $[PER_{end}]$*'s been convicted of* $[Crime_{start}]$ ***obstruction of justice*** $[Crime_{end}]$.

(3) $[Defendant_{start}]$ ***She*** $[Defendant_{end}]$*'s been convicted of* $[Crime_{start}]$ ***obstruction of justice*** $[Crime_{end}]$.

[14] We only used the input type representation and consider a complex output based on tokens, which is not considered in [97].

For the *Argument Role Markers*, if an entity has different roles in different events that are present in the same sentence, we mark the entity with all the argument roles that it has [98].

5.3.3 Transformer-based Question Answering Model

We formulate the ED task as a QA task, where, for every sentence, we ask if an event type of interest is present, and we expect a response with an event trigger, multiple event triggers, or none. Our model extends the BERT [3] pre-trained model, which is itself a stack of Transformer layers [2]. Differently from the Transformer-based classification models, a QA architecture is span-based, which means that instead of classifying every or as an event type, it detects the beginning and the end position of a sub-text in a text that could refer to a possible event trigger word.

To feed a QA task into BERT, we pack both the question and the reference text into the input, as illustrated in Figure 10. The question is regarding the existence of an event type in a text, and the response is the reference text from which the event trigger (response) is extracted. In Figure 10, the event trigger is *war*, representing an event of type *Attack*, and the QA model should extract this word from the reference text.

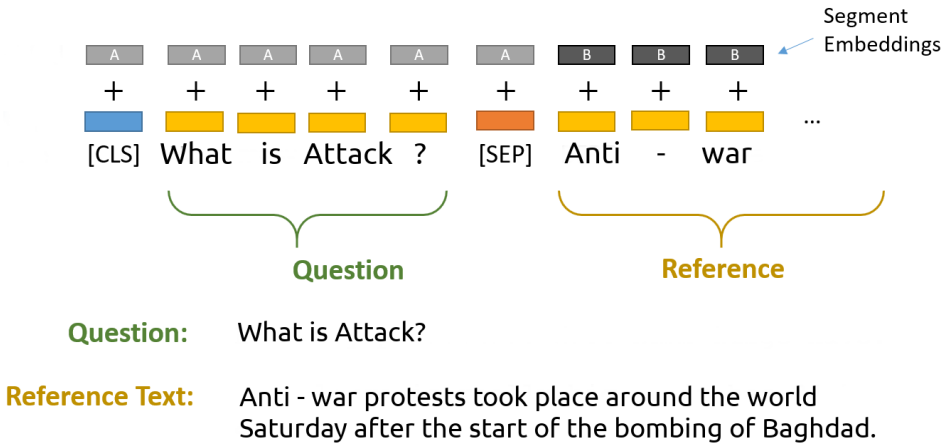

Figure 10: Example of input modification to fit the QA paradigm for a sentence that contains an event of type Attack.

Technically, the input embeddings are the sum of the token embeddings and the segment embeddings. The input is processed in the following manner: word/token embeddings (specifically to the BERT input and tokenisation [99], a pre-defined special token, [CLS], is added to the input word tokens at the beginning of the question and a [SEP] token is inserted at the end of both the question and the reference text) and segment embeddings (a marker indicating the question or the reference text is added to each token). This allows the model to distinguish between the question and the text [3].

To fine-tune BERT for a QA system and to detect the word of interest (event trigger), a start position vector and an end position vector are introduced. A linear layer is added at the top of BERT layers with two outputs for the start and end vectors of the answer. The probability of each word being the start or end word is calculated by taking a dot product between the final embedding of the word and the start or end vector, followed by a softmax over all the words. The word with the highest probability value is

considered. This method differs from the event detection approaches presented by [46] and [47] where the models are classification-based, instead of the span-based QA.

Next, for every type of event (*Demonstrate, Die, Attack*, etc.), we formulate the question by automatically generating them using the following template, where [Event Type] is replaced by each of the event types in the pre-set list:

What is the [Event Type] ?

An example of a sentence containing an *Attack* event is illustrated in Figure 10. All sentences in a document will be paired with each of the questions for all types of events. We also consider questions that do not have an answer in the case where an event of a specific type is not present in the sentence. When there is more than one event of the same type in a sentence, we consider that the question has multiple answers. From the n best-predicted answers, we consider all those that obtained a probability higher than a selected threshold.

The strategy of the threshold selection is represented by the Algorithm 1, an algorithm slightly similar to the method proposed by [46] for determining the number of arguments to be extracted for each role by finding a dynamic threshold. When the predicted chunks are self-contained as, for example, the noun chunks *assault* and *air assault* are predicted, we consider only the first predicted event trigger (*assault*).

Algorithm 1: Threshold selection for obtaining the top event triggers.

Input:

Development candidates $dev\_candidates$

Test candidates $test\_candidates$

$list\_thresh \leftarrow \{0.1, 0.2, 0.3, 0.4, 0.5\}$

$best\_thresh \leftarrow 0.0$

$best\_F1 \leftarrow 0.0$

for $thresh \in list\_thresh$ **do**

  $F1 \leftarrow eval(dev\_candidates)$

  if $F1 \geq$ *best_F1* **then**

    $best\_F1 \leftarrow F1$

    $best\_thresh \leftarrow thresh$

$final\_triggers \leftarrow \{\}$

for $candidate \in test\_candidates$ **do**

  if $candidate.probability \geq best\_thresh$ **then**

    $final\_triggers.add(candidate)$

Output: $final\_triggers$

Next, we also experiment with adding entity information, and for this, we utilised the model presented in Section 5.3.2.

5.4 Unsupervised Event Extraction with FrameNet

To accomplish the extraction of events in an unsupervised manner, we start by generating the dependency parse trees for each sentence in the dataset[15]. Next, we focus on the extraction of noun-phrases

[15] For this, we used spaCy 2.1+ [100].

(NPs) that can be pronouns, proper nouns or nouns, that are generally used as subjects (*nsubj*) or objects (*obj*) (or complements of prepositions). Finally, we obtain a triplet composed of the tree *root*, which is generally the verb of the sentence, and its dependents, the *nsubj* and the *obj*.

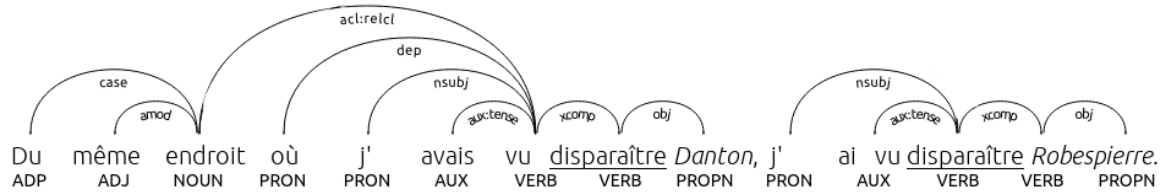

Figure 11: Example of the correspondence between syntactic arguments of the verbs and participants of the event denoted by the verb (translation: *From the same place where I had seen Danton disappear, I saw Robespierre disappear.*)

A candidate event mention is, thus, represented by a triplet, where the *root* is commonly a verb, which can possibly be mapped to a lexical unit (LU). For example: "disparaître" (to disappear) is a lexical unit for both events in Figure 11 and the $subj, obj$ are a set of syntactic dependencies associated to the LU (subject, object, etc.).

In Figure 11, the pronoun "je" (*I*) is the subject in both events, while the entities "Danton" and "Robespierre" are objects, with a set of possible semantic types (e.g. PER, LOC, etc.): "Danton" and "Robespierre" are two PER participants.

$$event_{candidate}(Life.Death) = \{root, subj, obj\}$$

Next, for each sentence, we pass it through a pre-trained language model[16] and we obtain a contextual representation for each token $x = [x_0, x_1 \dots x_n]$ where n is the sentence length. From this, we extract the *root* representation as an event candidate ($event_{candidate}$). For each event Type.Subtype mapped to FrameNet (i.e., *Conflict.Attack*), we generate a representation in the same manner. We first concatenate all the LUs for the Type.Subtype and pass it through the same pre-trained language model. We then extract the event Type.Subtype representation ($event(type)$).

For deciding the type of the candidate $event(type)$, we use cosine similarity for comparing the event representations[17], which is defined as

$$\cos(Event_{candidate}, LUs) = \frac{Event_{candidate} \times LUs}{\|Event_{candidate}\|\|LUs\|}$$

where $event_{candidate}$ is the vector representation of the event trigger candidate and LUs is the averaged representation of the concatenated n lexical units $\sum_{j=1}^{n} LU_j/n$.

For example, for *Attack*, we compare the extracted *roots* with the following set of lexical units that was retrieved from FrameNet: *attack, assault, strike, ambush, assail, raid, bomb, bombing, raid, infiltrate, hit, fire, small, take up arms, fire, airstrike, bombardment, counter-attack, counter-offensive*. After analyzing the results, we observed that two separate sets of event triggers were extracted:

[16] https://huggingface.co/bert-base-multilingual-uncased
[17] A threshold of 0.7 was chosen for considering the most similar event candidates.

- (1) known events: *foudroyer* (strike down), *armer* (take up arms), *attaquer* (attack), *frapper* (strike);
- (2) unseen events: *arracher* (snatch), *déchiqueter* (tear off), *étouffer* (suffocate), *empoigner* (grab), *trancher* (shred).

6 Experiments

In the NewsEye project, which focuses on historical newspapers, we are particularly keen on evaluating the performance of the models over texts that were the results of an automatic text recognition (ATR) process, as historical documents are evidently not digitally-born. The focal point of this set of experiments is to observe how the level of noise stemming from the digitisation process impacts the performance of the models. Thus, we perform several types of experiments depending on the dataset availability.

In order to create such an appropriate datasets, raw text from both datasets was extracted and converted into clean images. For the simulation of different levels of degradation on these images, we used DocCreator [101]. The rationale is to simulate what can be found in deteriorated documents due to time effect, poor printing materials or inaccurate scanning processes, which are common conditions in historical newspapers. We used four types of noise: *Character Degradation* adds small ink dots on characters to emulate the age effect on articles, *Phantom Character* appears when characters erode due to excessive use of documents, *Bleed Through* appears in double-paged document image scans where the content of the back side appears in the front side as interference, and *Blur* is a common degradation effect encountered during a typical digitisation process. After contaminating the corpus, all the text was extracted from noisy images using Tesseract optical character recognition (OCR) Engine v4.0[18] [102] to produce the digitised documents, for initial clean images (without any adulteration) and the noisy synthetic ones. An example with the degradation levels is illustrated in Figure 12.

The experiments were conducted in the following manner: for each noise type, the different intensity is generated to see its relation to the performance of the model. Character error rate (CER) and word error rate (WER) were calculated for each noise level, that can align long noisy text even with additional or missing text with the ground truth, thus enables it to calculate the error rate of OCR process.

The experiments are performed under conditions of varying word error rate (WER) and character error rate (CER):

- Original text (no OCR, 0% WER, 0% CER)
- OCR from high-quality text images ($\sim$1% WER, $\sim$0.5% CER)
- OCR on degraded text images synthetically produced with DocCreator (2$-$50% WER, 1$-$20% CER)

6.1 Evaluation Settings

For the evaluation of the performance of the event detection task, we use the standard metrics: *Precision* (P), *Recall* (R), and *F-measure* (F1), defined by the following equations:
$P = TruePositives/(TruePositives + FalsePositives)$

[18] https://github.com/tesseract-ocr/tesseract

Figure 12: Example of types of noise applied on ACE 2005 dataset: clean image, *Phantom Character*, *Character Degradation*, *Bleed Through*, *Blur*, and all mixed together.

$R = TruePositives/(TruePositives + FalseNegatives)$
$F1 = \frac{2 \cdot P \cdot R}{P+R}$

True positives are the samples classified as belonging correctly to a class. False negatives are classified as not belonging to a class, incorrectly. False positives are the samples classified as belonging to a class, incorrectly. Thus, precision is the fraction of relevant samples among the retrieved samples, while recall is the fraction of relevant samples that have been retrieved over the total amount of relevant samples. The F1 is the harmonic mean between these two. Because the data in ED tasks usually suffer from class imbalance, we compute the micro-averages of these metrics for aggregating the contributions of all classes.

For measuring the document distortion due to the OCR process, we also report the standard metrics: *character error rate* and *word error rate*.

Character error rate (CER) is defined as $CER = (i_c + s_c + d_c)/n_c$ where n_c is the ground truth in terms of character, i_c, s_c, and d_c are the number characters that needed to insert, substitute and delete respectively to reconstruct the transcribed text into the ground-truth.

Similarly, *Word Error Rate* (WER) is calculated as $WER = (i_w + s_w + d_w)/n_w$ where all the parameters

remain the same, except they are counted in words. It is worth noting that WER is generally higher than CER within the same sample, as WER is a stricter evaluation where any character mistake would make a whole word considered as wrong. On the other hand, CER is not as tight as the fore-mention, since the error in character is independent of each other and does not affect any previous or subsequent characters.

6.2 Experiments on DAnIEL dataset

For the purpose of comparing the two approaches, the data with a total of 4,822 documents was split at document level, 3,857 documents for training (80%), 482 documents for validation (10%), and the rest of 483 documents for testing (10%), stratified by language, as shown in Table 3.

	Total documents	Polish	Chinese	Russian	Greek	French	English
Train	3,857 (377)	281 (22)	357 (13)	341 (28)	312 (16)	2,186 (269)	380 (29)
Validation	483 (51)	35 (3)	45 (2)	42 (6)	39 (5)	274 (33)	48 (2)
Test	482 (61)	36 (5)	44 (1)	43 (7)	39 (6)	273 (38)	47 (4)

Table 3: DAnIEL dataset splits. In (relevant), the number of documents annotated with events is reported.

Hyperparameters CNN-based model For the CNN-based model, the set of parameters is as it follows. The filter sizes for the convolutional layer with $tanh$ activation are from the set $\{1, 2, 3\}$ to generate feature maps, in order to represent n-grams, each of them with size 300. The window size for triggers is set to 31 while the dimensionality of the position embeddings is 50. Two dropouts are applied, one dropout rate of 0.5 after the layer for embeddings words, and a dropout rate of 0.3 before $softmax$, after the max-over-time pool operation. The batch size is 256. For these experiments, we use no pretrained embeddings (due to the multilingual corpus, challenge that we are not approaching in these experiments), they are initialised based on a normal distribution by default, leaving the opportunity that the embeddings are learnt on the task.

The regularisation is implemented by early stopping [103], with a patience of 3 epochs, consisting in stopping the training as soon as the error on the validation set is higher than it was in the previous epoch. Training is done via stochastic gradient descent with shuffled mini-batches and the *AdaDelta* [104] update rule. During the training, the word and positional embeddings tables are optimised at the same time.

Hyperparameters DAnIEL For DAnIEL, the ratio is set to 0.8.

Evaluation framework We perform one type of evaluation, at the document level (specific DAnIEL): a document represents an event if the triggers are correctly found and match with the gold-truth ones.

6.2.1 Experiments with Clean Data

		Polish	Chinese	Russian	Greek	French	English	All
DAnIEL (%)	P	30	50	25	58.33	50.48	40	42.35
	R	25	50	28.57	53.85	44.17	40	46.15
	F1	27.27	**50**	**26.67**	**56**	47.11	40	44.17
CNN-based (%)	P	100	0	66.67	50	60.23	75	60.75
	R	16.67	0	14.29	38.46	50.48	30	41.67
	F1	**28.57**	0	23.53	43.48	**54.92**	**42.86**	**49.43**

Table 4: Evaluation of the CNN-based model and DAnIEL on the initial test data for event detection.

We can observe from Table 4, that DAnIEL is more balanced regarding the precision and recall metrics, being able to have higher F1 on the under-represented languages (Chinese, Russian, and Greek) than the CNN-based model.

The CNN-based model struggles with the recall values. This might be related to the fact that the model is not able to detect some locations due to the fact they are not mentioned in the original text, whether DAnIEL is capable tu use external resources and article metadata in order for them to be inferred. The DAnIEL system is able to detect correctly only the disease, but the CNN-based model cannot retrieve any of them correctly, even more, the location. Besides this, the small amount of data greatly affects the performance of the CNN-based model. We assume that the CNN-based model performs better for the French documents, due to the larger amount of data, and for the English documents, due to the fact that, in the annotation process, all the disease−location pairs (in the English documents, no number of victims was annotated) were located in the texts, and thus a higher chance of better performance.

Finally, the CNN-based performed slightly better in total than DAnIEL, with a difference of 5.26 percentage points in F1. We add also that one issue that needs to be further studied is DAnIEL's false positives problem: for instance documents relating vaccination campaigns are usually tagged as non-relevant in the ground truth dataset.

Further experiments have been performed on the clean data with Transformer-based models. We chose as the pre-trained models the BERT-multilingual cased and uncased models due to multilingual nature of the dataset. The results are presented in Table 5 and they prove a considerable improvement of around 50% comparing with the previous systems in regards to the F1-score.

Pre-trained Models	Polish	Chinese	Russian	Greek	French	English	All
BERT-multilingual-cased	68.9	71.7	0	56.41	22.86	82.35	50.37
BERT-multilingual-uncased	69.25	76.92	0	61.11	22.22	85.71	52.53

Table 5: F1 scores of the Transformer-based model and DAnIEL on the initial test data for event detection.

		Clean	CharDeg	Bleed	Blur	Phantom	All
All	CER	2.61	9.55	2.83	8.76	2.65	11.07
	WER	4.23	26.23	5.93	19.05	4.71	27.36
Polish	CER	0.15	5.86	0.19	7.57	0.19	5.51
	WER	0.74	20.66	1.17	13.23	1.17	20.70
Chinese	CER	36.89	41.01	38.24	43.97	36.91	46.97
	WER	–	–	–	–	–	–
Russian	CER	0.93	16.20	1.45	8.13	1.03	10.91
	WER	1.63	28.46	6.61	14.94	2.73	29.72
Greek	CER	3.52	9.04	3.76	13.79	3.54	16.28
	WER	15.86	41.36	17.39	54.02	15.93	54.76
French	CER	1.96	8.37	2.13	7.43	2.0	10.90
	WER	3.33	23.56	4.89	16.31	3.76	26.07
English	CER	0.35	5.75	0.52	4.74	0.44	7.43
	WER	0.66	24.78	2.14	14.72	1.66	20.99

Table 6: Document degradation OCR evaluation on the DAnIEL dataset.

6.2.2 Experiments with Noisy Data

The results in Table 6 clearly state that *Character Degradation* is the effect that affects the most the transcription of the documents. However, for character-based languages (e.g., Chinese), CER is commonly used instead of WER as the measure for OCR, and, thus, we report only the CER [105].

Also, regarding the Chinese documents, the high values for CER, for every type of noise, might be caused by the existence of the enormous number of characters in the alphabet that, by adding such an effect as *Character Degradation* can change drastically the recognition of a character (and in Chinese, one single character can often be a word). Otherwise, while *Character Degradation* noise and *Blur* effect have more impact on the performance of DAnIEL than *Phantom Character* type since it did not generate enough distortion to the images. A similar case applies for the *Bleed Through* noise.

Next, we present the results for event identification and classification for both systems. Results indicated in bold are the best F1 scores given by the system according to the type of degradation. We compute also a δ measure that gives the minimum decrease rate between the F1 given using clean data and the F1 given using noisy data for each type of degradation. This measure represents the perfect system which will give the best F1 for all degradation levels. We also present the evolution of the δ measure according to the types of noise, for both systems, for each language.

Regarding the experiments with the DAnIEL system, from the Table 7, we notice, first of all, that the *Character Degradation* effect, *Blur*, and most of all, all the effects mixed together, have indeed an impact or effect over the performance of DAnIEL, but with little variability. Meanwhile, *Phantom Degradation* and *Bleed through* had very little to no impact on the quality of detection with DAnIEL.

The cause of the decrease in performance of DAnIEL is that to detect events, it looks for repeated substrings at salient zones. In the case of many incorrectly recognised words during the OCR process, there may be no repetition anymore, implying that the event will not be detected. However, since DAnIEL only needs two occurrences of its clues (substring of a disease name and substring of a location), it is

assumed to be robust to the loss of many repetitions, as long as two repetitions remain in salient zones.

Language	Original	Clean	CharDeg	Bleed	Blur	Phantom	All
All	44.17	48.28	43.94	**48.61**	45.59	48.28	45.78 ↑
Polish	27.27	20	12.5	27.27	20	20	**36.36** ↑
Chinese	**50**	**50**	**50**	**50**	**50**	**50**	**50**
Russian	26.67	31.25	**33.33**	**33.33**	20	31.25	30.77↑
Greek	**56**	40	19.05	40	11.76	40	31.58 ↓
French	47.11	55.61	53.11	55.61	**58.38**	55.61	53.09 ↑
English	**40**	**40**	25	**40**	12.5	**40**	25 ↓

Table 7: Evaluation of DAnIEL results on the noisy test data for event detection.

For all the languages, Figure 13, the error rare (δ) can exceed 5% when using noisy data, with WER and CER reaching more than 9 and 25 respectively, relatively high degradation values. Thus, the event detection performance scores are generally directly increasing along with the level of the digitisation errors.

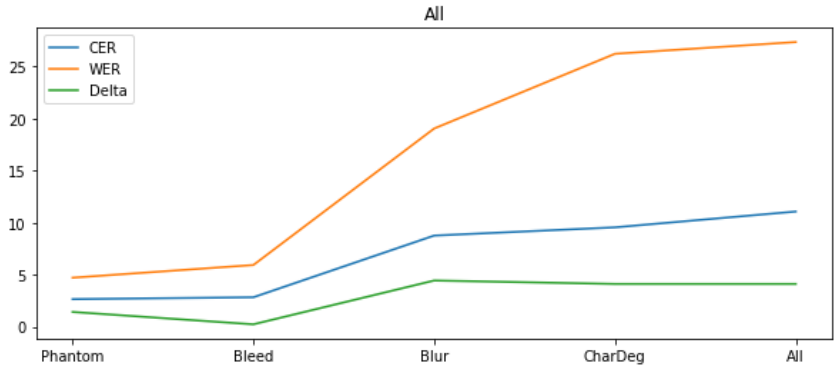

Figure 13: F1 degradation according to OCR error rates for event detection for the DAnIEL system, for all the languages.

Language	Original	Clean	CharDeg	Bleed	Blur	Phantom	All
All	**49.43**	47.62	37.21	46.47	40.38	47.62	38.28 ↓
Polish	**28.57**	15.38	0	**28.57**	15.38	15.38	0 ↓
Chinese	0	0	0	0	0	0	0
Russian	23.53	23.53	23.53	13.33	**25**	23.53	**25** ↑
Greek	**43.48**	33.33	14.29	26.67	0	33.33	0 ↓
French	54.92	**55.5**	45.28	54.35	50.31	**55.5**	48.72 ↓
English	42.86	18.18	**100**	18.18	0	18.18	0 ↓

Table 8: Evaluation results of the CNN-based model on the noisy test data for event detection.

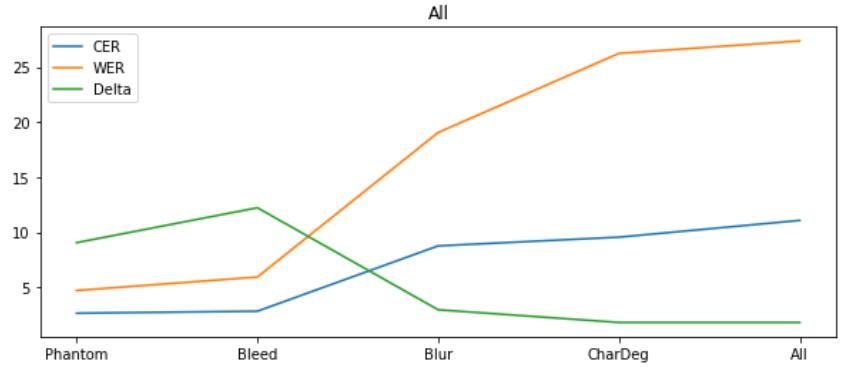

Figure 14: F1 degradation according to OCR error rates for event identification and classification for the CNN-based model, for all the languages

Table 8 analyses the effect of applying noise on the document images for the CNN-based model. The decrease in precision and recall is produced similar to the DAnIEL system, the impact on the scores being higher for the *Character Degradation*, *Blur*, and all mixed together, also. One drawback of this model is that it is based on embeddings at word-level, which can degrade the performance in the case of many modified words in the test set during the OCR process.and CER reaching more than 9 and 25 respectively.

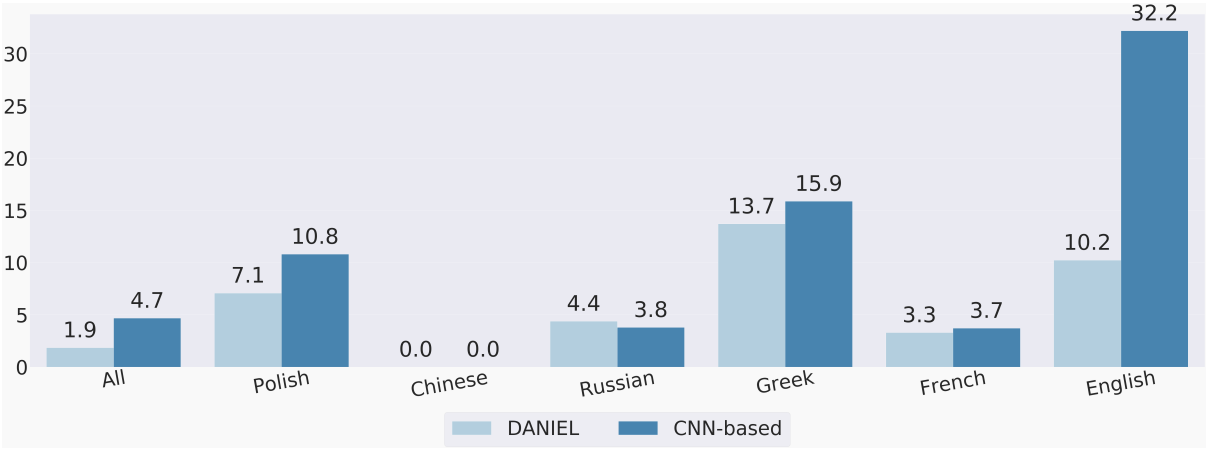

Figure 15: Standard deviations of the F1-scores for all effects mixed together, per language, event detection.

Studying the degree of variability of F1-scores for all the effects mixed together for event identification, and for event identification and classification, we notice the CNN-based model is more sensitive to the added effects, as shown in Figure 15. We conclude that using representations at word-level in the CNN-based model indeed hurts the performance of the model when evaluated on the text transcribed from degraded images.

Regarding all the results aforementioned, for the DAnIEL system, and the CNN-based model, computing the number of affected event words (disease, location, number of patients), we also notice that a very small number of them have been modified by the OCR process, only 1.98% for all the languages together, for all the effects mixed together, not far from the 1.63% that were affected by the OCR on

clean data. This is due to the fact that DAnIEL dataset is highly imbalanced (only 10.14% of a total of $4,822$ documents contain events), and it brings us to the conclusion that the event detection task is not considerably impacted by the degradation of the image documents.

One interesting observation is that the precision or the recall can increase, resulting in a higher F1, despite the higher noise effect applied, for event classification with the CNN-based model, where the δ is decreasing when applying all the degradation types. From our observation, it is because of that with a greater level of noise some false positives disappear. Documents, which were previously wrongly classified due to being too ambiguous to the system (for instance, as mentioned before, documents related to vaccination campaigns are usually annotated as irrelevant in DAnIEL dataset, were given much more distinction due to the noise, thus making them look less like relevant samples to the system.

This may seem counter-intuitive but noise can improve classification results, see for instance [106] for a study on the same dataset of the influence of boilerplate removal on results.

6.3 Experiments on ACE 2005 dataset

For comparison purposes, we use the same test set with 40 newswire articles (672 sentences), the same development set with 30 other documents (863 sentences) and the same training set with the remaining 529 documents (14,849 sentences) as in previous studies of this dataset [27, 29, 32, 35, 36]. We perform the following evaluation from the ACE 2005 evaluation: a trigger is correct if its event subtype and offsets match those of a reference trigger.

6.3.1 Experiments with Clean Data

We first consider four baselines based on the BERT language model, applied similarly to [107] for the named entity recognition (NER) task, with the recommended hyperparameters. We test four widely used pre-trained English language models, two based on BERT-base and two based on BERT-large, *cased* (trained on the original words) and *uncased* (trained on lowercased words).

We compare our proposed models with markers with several state-of-the-art neural-based models proposed for event detection, that do not use external resources, more specifically with the following models based on CNNs and RNNs: the CNN-based model [35] with and without the addition of gold-standard entities, and the recent proposed BERT-based models, the fine-tuned baseline BERT-base-uncased [46], the QA-BERT [46] where the task has been approached as a question answering task, the two models with adversarial training for weakly supervised event detection [68], the fine-tuned baseline BERT-base-uncased [46], the BERT_QA_Trigger [46], and the RCEE_ER (Reading Comprehension for Event Extraction, with *ER* that denotes that the model has golden entity refinement) [47], and the BERT and LSTMs approaches [69] that models text spans and captures within-sentence and cross-sentence context.

	Precision	Recall	F1-score
State of the Art Approaches			
CNN [35]	71.9	63.8	67.6
CNN [35]+	71.8	66.4	69.0
`BERT-base-uncased` & LSTM [69]	N/A	N/A	68.9
`BERT-base-uncased` [69]	N/A	N/A	69.7
`BERT-base-uncased` [46]	67.1	73.2	70.0
BERT_QA_Trigger [46]	71.1	73.7	72.3
DMBERT [68]	77.6	71.8	74.6
RCEE_ER [47]$^+$	75.6	74.2	74.9
DMBERT+Boot [68]	77.9	72.5	75.1
Proposed Models			
CNN-based Models			
Our CNN (replicated, changed hyperparameters)	68.8	66.1	67.4
Transformer-based Classification Models			
`BERT-base-uncased`	71.6	68.4	70.0
`BERT-base-cased`	71.3	72.0	71.6
`BERT-large-uncased`	72.0	72.9	72.5
`BERT-large-cased`	69.3	77.1	73.0
Transformer-based Classification Models with Entities			
`BERT-large-cased`+*Entity Position Markers*$^+$	75.9	76.6	76.2*
`BERT-large-cased`+*Entity Type Markers*$^+$	79.3	77.8	**78.5***
`BERT-large-cased`+*Argument Role Markers*$^+$	78.9	80.4	79.6*
Transformer-based QA Models			
`BERT-QA-cased-squad2`	69.6	68.1	68.9
`BERT-QA-uncased-squad2`	70.6	66.7	68.6
`BERT-QA-cased`	62.2	74.3	67.7
`BERT-QA-uncased`	68.4	70.5	69.4
Transformer-based QA Models with Entities			
`BERT-QA-base-cased` + *Entity Position Markers*$^+$	74.9	72.4	73.6*
`BERT-QA-base-cased` + *Entity Type Markers*$^+$	76.3	72.2	74.2
`BERT-QA-base-uncased` + *Entity Position Markers*$^+$	78.0	70.7	74.2*
`BERT-QA-base-uncased` + *Entity Type Markers*$^+$	78.5	77.2	**77.8***
`BERT-QA-base-cased` + *Argument Role Markers*$^+$	79.8	75.0	77.3*
`BERT-QA-base-uncased` + *Argument Role Markers*$^+$	83.2	80.5	81.8*

Table 9: Evaluation of our models and comparison with state-of-the-art systems for event detection on the blind test data. $^+$ with gold entities or arguments. Statistical significance is measured with McNemar's test. * denotes a significant improvement over the previous model at $p \leq 0.01$.

Between the BERT-based baseline models presented in Table 9, it is worth noticing that the *cased* models perform better than the *uncased* ones, which could confirm that named entities that are usually capitalized are an important clue for the event detection task[19]. Moreover, the results are similar to the BERT-base-uncased in [46] (the same F1 value and similar precision and recall scores) and [69]

Full results of our model and its comparison against state of the art is presented in Table 9. There is a

[19] An amount of around 30% of the entities and 3% of the event triggers have the first token capitalized.

significant gain with the trigger classification of 9.04% higher over the stand-alone BERT-based model and 5.99% to the best reported previous models. These results demonstrate the effectiveness of our method to incorporate the argument information.

Moreover, the improvements are consistent regardless of the type of encoder (BERT or other) used to represent the inputs. For our first model (*Entity Position Markers*), where the entities are surrounded by a general marker that does not depend on the entity type, the results are improved with three percentage points revealing that the position of the entities is relevant for the trigger detection task. Furthermore, when we mark the entities with their argument roles (*Argument Role Markers*), the recall and F1 increase with around one absolute percentage point. However, this case is substantially optimistic as it assumes that argument roles were correctly identified and typed.

We first experiment with several BERT-based pre-trained models. First, we consider the models trained on SQuAD 2.0 tailored for the extractive QA downstream task (base-cased-squad2[20] and bert-uncased-squad2[21]), and the more general BERT models trained on large amounts of data and frequently used in research. The results are reported in Table 9, where we can easily observe that the *BERT-base-uncased* obtains the highest values. The *large* BERT-based models were not considered due to memory constraints[22]. We also distinguish between the *cased* and *uncased* models, where the *squad2* F1 values are marginally close, as well as for the *base* models.

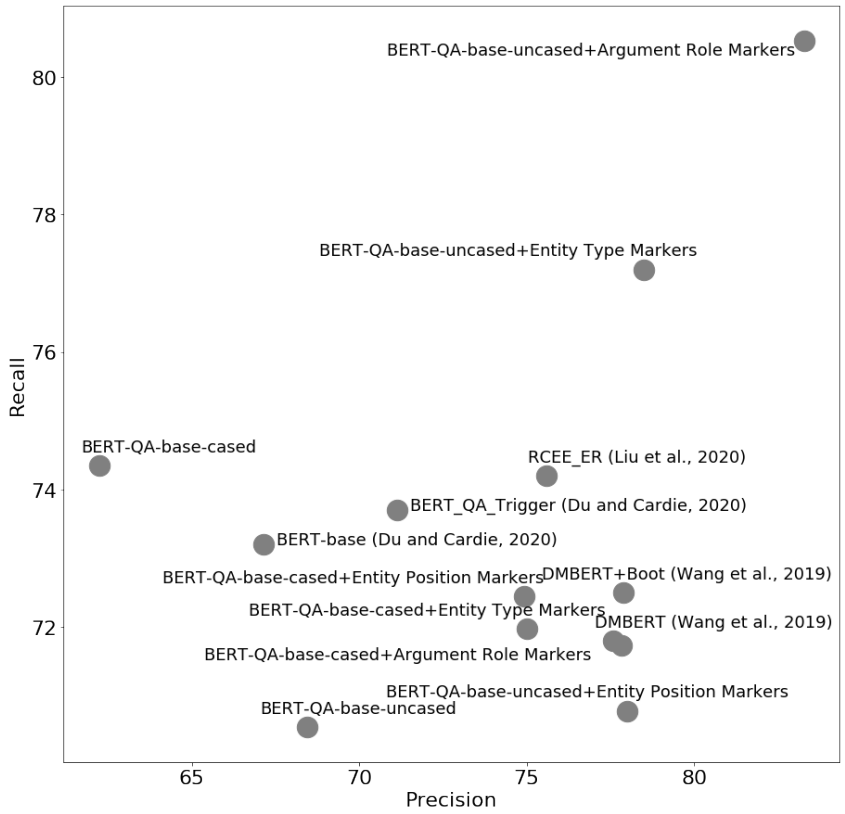

Figure 16: Precision versus recall for the state-of-the-art models and our proposed approaches.

[20] https://huggingface.co/deepset/bert-base-cased-squad2

[21] https://huggingface.co/twmkn9/bert-base-uncased-squad2

[22] Reducing the size of some hyperparameters for the *large* models, as well as reducing the size of the batch size, decreased considerably the performance. The F1 value even plateaued at 0%. We ran the models on a machine with four GeForce RTX 2080 GPUs, with 11,019 MiB each.

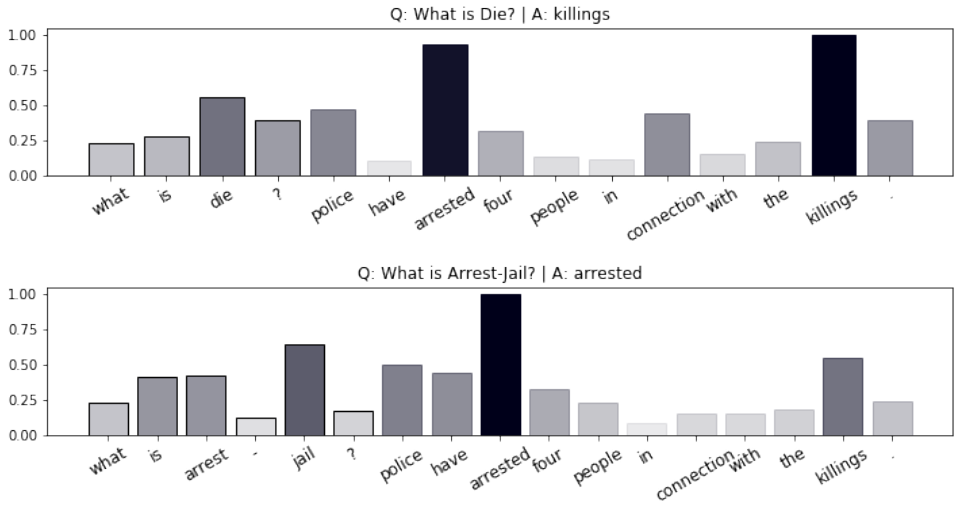

Figure 17: An example of a sentence that contains two events: *Die* event triggered by the word **killings** and *Arrest-Jail* event triggered by **arrested**. The model used is BERT-QA-base-uncased.

In Table 9, we present the comparison between our model and the latest state-of-the-art approaches[23].

When compared with the BERT_QA_Trigger [46], our models that use either the positions or the types of the entities bring a considerable improvement in the performance of trigger detection. It is clear that further marking the entities with their types can increase both precision and recall, balancing the final scores.

From Figure 16, one can observe that, on the diagonal, the most balanced models with regard to the precision and the recall values mainly consist in the models that include either *Entity Type* or *Argument Role Markers*, along with the QA classification-based models proposed by [46, 47].

For the BERT-QA-base-uncased + *Entity Position Markers* model and the DMBERT-based models [68], the results are visibly imbalanced, with high precision values, which implies that these models are more confident in the triggers that were retrieved. Moreover, marking the entities with <E> and </E> and BERT-QA-base-uncased has the lowest values in recall, but adding *Position Markers* clearly increases the precision of the results.

While entities can be present in the entire document, arguments can only surround event triggers. Knowing the argument roles beforehand brings further improvements, and we assume that an important reason for this is that, since the arguments are present only around event triggers, this could help the language model to be more aware of the existence of an event or multiple events in a sentence.

Even though the models that integrate argument roles (`BERT` pre-trained language models + *Argument Role Markers*) have the highest performance, we cannot necessarily compare the models that utilise the entities. This is due to the fact that, while entities can be present in the whole document, they do not necessarily to participate in an event. Thus, the task of detecting entities can be treated as a named entity recognition (NER), an independent task that can be performed before. Argument role detection task has an increased level of difficulty because, generally, it depends on the event trigger detection. We

[23] While there are several works that rely on gold entity types, due to the lack of space, we only considered the ones based on pre-trained language models.

consider, therefore, in a more realistic scenario, that the detection of entities and their inclusion in the model for event detection is a more practical and pragmatical solution for increasing the performance.

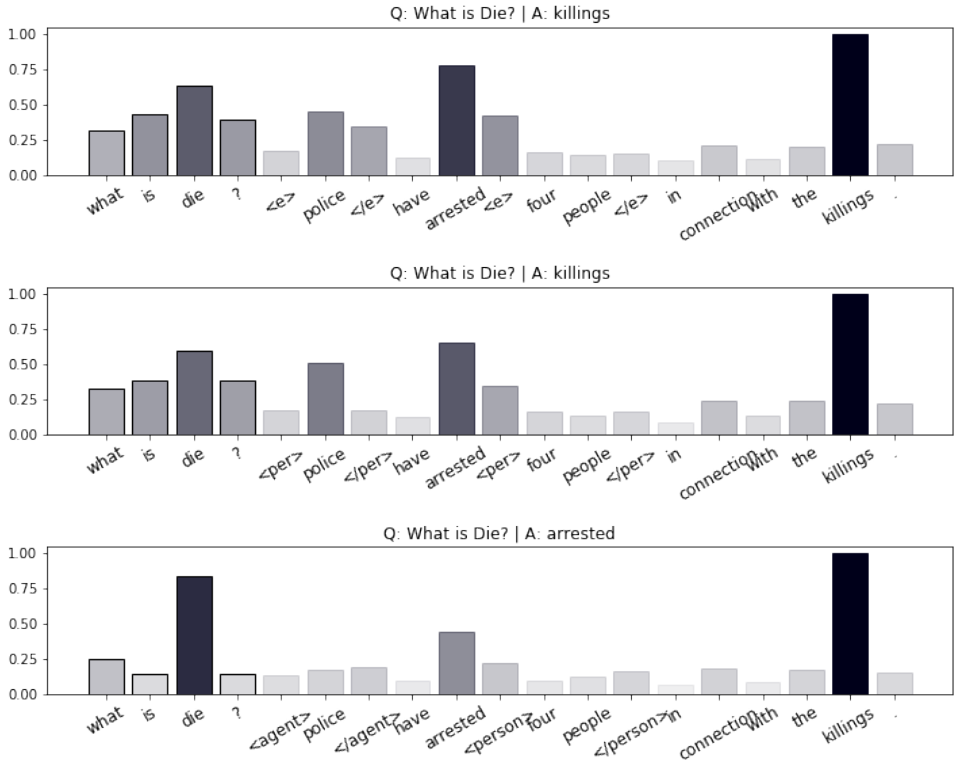

Figure 18: An example for the *Die* event triggered by **killings** with three types of markers: *Entity Position, Entity Type*, and *Argument Role Markers*.

Figure 19: [CLS] representation of each sentence in the test set that contains at least an event for BERT-QA-base-uncased, BERT-QA-base-uncased + *Entity Position Markers+*, BERT-QA-base-uncased + *Entity Type Markers+*, and BERT-QA-base-uncased + *Argument Role Markers*.

For a deeper analysis of the impact of entity information, we leverage the gradients in our proposed models to efficiently infer the relationship between the question, context, and the output response. [108] studied the identifiability of attention weights and token embeddings in Transformer-based models. They show that the self-attention distributions are not directly interpretable and suggest that simple gradient explanations are stable and faithful to the model and data generating process. Thus, as applied by [109], to get a better idea of how well each model memorizes and uses memory for contextual understanding,

we analyze the connectivity between the desired output and the input. This is calculated as:

$$\text{connectivity}(t, \tilde{t}) = \left|\left| \frac{\partial y_k^{\tilde{t}}}{\partial x^t} \right|\right|_2$$

where t is the time index, $\tilde{t}$ the output time index, and the result is the magnitude of the gradient between the logits for the desired output $y_k^{\tilde{t}}$ and the input x^t. The connectivity is computed with respect to both start position and end position of the answer, then it is normalized, and it is visible as saliency maps for every word in Figures 17 and 18[24].

By looking at the gradients in Figure 17, where two events of different types are present, we can observe, in the upper part of the figure, that while the model sees the word **killings** and **arrested** as impactful, it also sees the words *police, connection* as impactful and selects an answer in that neighborhood. Even though both trigger candidates **killings** and **arrested** have a clear impact due to their gradient values, by looking at the probability values, **killings** is recognized with a 99.4% probability, while **arrested** obtained a probability of 2.3×10^{-7}, value that is lower than our selected threshold 0.2. In the lower part of the figure, for the question *What is Arrest-Jail?*, the words *die, police,* **killings** clearly influence the choice of the answer **arrested**.

In Figure 18, we present the same sentence with the three types of input modifications: *Entity Position Markers, Entity Type Markers*, and *Argument Role Markers*, with the *What is Die?* question and the correct answer **killings**. In the upper part of the figure, where the sentence has been augmented with the entity position markers <E> and </E>, we notice that the words that impact the most the result are **killings** along with *die*, **arrested**, and *police*. In this case, one can also see that the end marker </E> contributed too.

In the middle part of the figure, where the sentence has been augmented with the entity position markers <PER> and </PER> for the two entities *police* and *four people*, the influence of other words as in *die*, **arrested**, and *police* slightly decreased. In the bottom part of the image, the gradients of these words are visibly reduced.

When the sentence is augmented with argument roles, <Agent>, </Agent>, <Person> and </Person>, the noise around the correct answer has noticeably diminished, being reduced by the additional markers. The most impactful remaining words are the word *die* in the question and the correct answer **killings**.

In order to analyze the quality of the sentence representations, we extract the [CLS] representation of each sentence for BERT-QA-base-uncased and for BERT-QA-base-uncased + *Argument Role Markers*. Then, we plot these representations in two spaces where the labels (colors of the dots) are the event types, as illustrated in Figure 19. On the right-hand side of the figure, where argument role markers are used, it is clear that the sentence representations clusters are more cohesive than when no entity information is considered (left-hand side), thus confirming our assumption regarding the importance of the entity informative features in a QA system.

Evaluation on Unseen Event Types We follow the same strategy as [46] where we keep 80% of event types (27) in the training set and 20% (6) unseen event types in the test set. More exactly, the unseen event types were chosen randomly and they are: *Marry, Trial-Hearing, Arrest-Jail, Acquit, Attack*, and

[24] The sentence is lowercased for the *uncased* models.

Declare-Bankruptcy. Table 10 presents the performance scores of our models for the unseen event types.

We compare with BERT-QA-Baseline which is our baseline that selects an event trigger in a sentence without being trained on ACE 2005 data. Since the models proposed for ED by [46] and [47] are classification-based in a sequential manner, they are not capable of handling unseen event types.

From the results, without any event annotation, the BERT-QA-base-uncased-Baseline obtains a low F1 value (1.38%). We observe that the performance values increase proportionally to the specificity of the markers. Thus, it is not surprising that the highest values are obtained when the argument roles are marked, also obtaining the highest precision. These results also confirm the effectiveness of our proposed models in handling unseen event types.

Approaches	**P**	**R**	**F1**
BERT-QA-base-uncased-Baseline	0.75	8.33	1.38
BERT-QA-base-uncased	47.75	26.76	31.17
BERT-QA-base-uncased + *Entity Position Markers*$^+$	44.02	47.58	37.39
BERT-QA-base-uncased + *Entity Type Markers*$^+$	53.61	**54.43**	50.43
BERT-QA-base-uncased + *Argument Role Markers*$^+$	**83.33**	47.40	**53.64**

Table 10: Evaluation of our models on unseen event types. $^+$ with gold entities or arguments.

6.3.2 Experiments with Noisy Data

For a first experiment, we explored the ability of the CNN-based model of handling noisy data, more exactly, on the synthetically created data.

	Original	Clean	CharDeg	Bleed	Blur	Phantom	All
CER	0	0.83	4.10	1.34	7.28	0.95	14.81
WER	0	1.13	17.96	5.61	18.49	2.50	35.93
Affected triggers	0	0.94	19.05	2.11	19.05	0.94	41.17

Table 11: Evaluation results on the noisy test data for event identification + classification. CharDeg = character degradation, Bleed = Bleed through, All = CharDeg + Bleed + Phantom + Blur

Table 12 illustrates the effect of applying noise on the document images for the CNN-based model. The decrease in precision and recall is produced similar to the DAnIEL, the impact on the scores being higher for the *Character Degradation*, *Blur*, and all mixed together, also. We recall that one drawback of this model is that it is based on a pre-defined set of word embeddings, which can degrade the performance in the case of many wrongly detected words in the OCR process. The results, however, are consistent with the drop in the quality of the documents, and thus, for the two highest values of CER, 4.10 for *Character Degradation* and 14.81 for all the noise effects together, the lowest F1 values were obtained, 48.97, and 40.77 respectively.

In a deeper analysis, we observed that the number of event triggers that were affected by the OCR process when all the noise levels were applied is 41.17% out of all event triggers, and thus, this justifies the large drop of around 27 percentage points. Also, while 19.05% of the event triggers were affected in two cases, *Character Degradation* and *Blur*, the CER error rates (4.10 and 7.28, respectively) and the F1 values differ (48.97% and 59.50% respectively). An explanation is that the precision of the results in the case of the *Blur* is considerably higher than in the case of *Character Degradation*, which would mean that even though both models managed to retrieve a similar amount of event triggers (a recall of 50.54% and a recall of 53.77%), the CNN-based models were able to better detect the correct event type even when the words was affected by the *Character Degradation* noise. When comparing with the digitisation impact onto the DANIEL-DATA, we conclude that the imbalanced nature of both datasets is a factor in assessing the level of impact that the digitisation process can have on the ED task. DANIEL-DATA is highly imbalanced, with only 10.14% of a total of $4,822$ documents contain a disease name and a location, while, in the case of the ACE 2005 dataset, 92.32% of the documents are contain events (generally multiple events). Thus, this could further explain the higher impact of these type of errors on the performance of event detection in ACE 2005.

	Original	Clean	CharDeg	Bleed	Blur	Phantom	All
P	68.82	68.62	47.63	57.75	67.55	59.05	48.02 ↓
R	66.13	65.51	50.54	64.37	53.77	64.94	35.48 ↓
F1	67.40	66.97	48.97	60.82	59.80	61.72	40.77 ↓
CER	0	0.83	4.10	1.34	7.28	0.95	14.81
WER	0	1.13	17.96	5.61	18.49	2.50	35.93

Table 12: Evaluation results on the noisy test data for event detection. CharDeg = character degradation, Bleed = Bleed through, All = CharDeg + Bleed + Phantom + Blur

Analysing the results, we notice that for all the noise effects together, 41.17% of the trigger words were affected as shown in Table 11, which is a large amount of event triggers, and for this reason, a large drop in performance of almost 27 percentage points in F1.

We consider that the Transformer-based models would, in this case, be more efficient in handling noisy data, and thus, we evaluate these models in a real-case scenario, in the context of NewsEye project.

7 Evaluation on NewsEye Selected Subsets

The difficulty of detecting events in the NewsEye dataset does not only refer to the ATR or digitisation errors, but also to the lack of annotated data in a multilingual setting. Thus, we decided to annotate a few documents regarding previously chosen subjects, and to experiment with our best event detection systems in a domain and language adaptation scenario.

7.1 Data Collection and Annotation

The documents were collected using the NewsEye platform [110], and annotated by the Digital Humanities groups from the NewsEye consortium from the University of Innsbruck (UIBK-ICH), Austria, and the

Paul Valéry University, France. The sujects of the datasets were selected by the annotators, depending on their line of research and interests. The articles were selected by using the NewsEye Demonstrator. This process implies the training of our models on the English ACE 2005 dataset with zero-shot learning. There were 25 German documents and 8 French document annotated in order to assess the ability of our model for, not only domain and language adaptation, but also noisy articles due to ATR.

7.2 Evaluation Settings

For this evaluation setting, we did not consider the DAnIEL system due to its specificity in detecting only epidemic-related events, nor the CNN-based model, since the performance of such system was the lowest in our preliminary experiments.

We fine-tuned a pre-trained multilingual BERT model on monolingual English ACE 2005 [25]. The BERT authors published multilingual pre-trained models in which the tokens from different languages share an embedding space and a single encoder model. In the same manner, for detecting events in a multilingual setting, we utilise a zero-shot configuration. This means that the pre-trained multilingual BERT system is fine-tuned on English data, and then evaluated on the foreign languages in NewsEye.

In order to evaluate our models using this configuration, we chose the best performing of our models, the Transformer-based classification models with entity markers. We could not report the results for the architectures with the question answering paradigm, as they are span-based and thus, language-wise, a model trained with English tokenisation and applied on other languages reflected the importance of the different linguistically meaningful units from the surface text[26]. The entities were predicted with the methods utilised currently in the NewsEye platform, methods based on a pre-trained and finetuned language models with a stack of Transformer encoders on top in order to alleviate the digitisation errors by capturing character-level information [12, 111].

7.2.1 Evaluation on a French NewsEye Subset: Women's Right to Vote

Regarding the French dataset annotated with events, the different articles from the chosen sample were particularly interesting for identifying the different types of events organised around women's right to vote. While some activist meetings organised by the suffragettes are being prepared, protests led by organisations are also mentioned. These few articles already allow us to identify not only important entities for the theme of female suffrage, but also the relations and events between these entities. The number of annotated events are presented in Figure 21. To understand the meaning of the event types present in these articles in a deeper analysis, we detail several types in the following paragraphs.

The *Meet* subtype belongs to the *Contact* event type. A *Meet* event occurs whenever two or more entities come together at a single location and interact with one another face-to-face. *Meet* events include talks, summits, conferences, meetings, and any other event where two or more parties get together at some location. However, they can be easily confused with the *Transport* event subtype. The *Transport* subtype belongs to the *Movement* event type. A *Transport* event occurs whenever an artifact (weapon or vehicle) or a person is moved or travels from one place to another. Thus, if someone travels to a location, to have a meeting, this is generally a *Transport* event. An example of a *Meet* event form

[25] This approach proved to be efficient for named entity recognition [111].
[26] Preliminary experiments proved very low performance.

the NewsEye dataset is: *C'est ainsi qu'aujourd'hui à Hyde-Park, deux réunions de suffragettes (constitutionnelles) ont pu avoir lieu sans aucune interruption. (Thus today in Hyde-Park, two meetings of suffragettes (constitutional) were able to take place without any interruption.)* where réunions (meetings) triggers an event of type *Meet*.

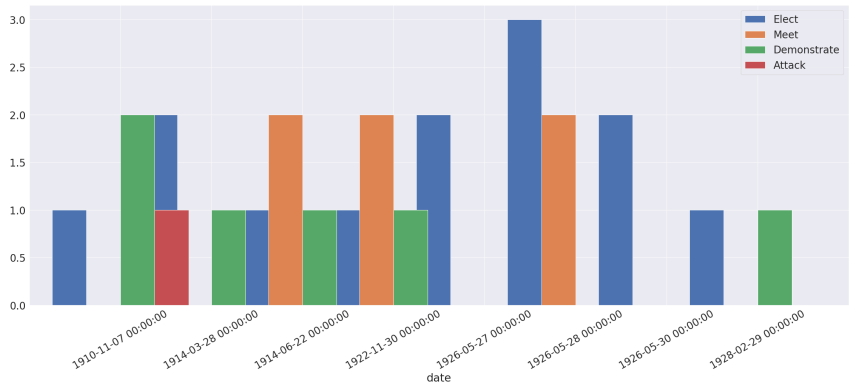

Figure 20: The number of annotated events in French.

Event Type	P	R	F1
`BERT-multilingual-uncased`			
Demonstrate	80	66.7	72.7
Elect	50	7.7	13.3
Meet	100	100	100
All	76.6	58.1	**62**
`BERT-multilingual-uncased` + *Entity Type Markers*$^{+}$			
Demonstrate	100	33.3	50
Elect	75	23.1	35.3
Meet	100	83.3	90.9
All	91.6	46.5	57.5

Table 13: Evaluation of NewsEye French event detection.

An *Elect* event occurs whenever a candidate wins or participates in an election designed to determine a person argument of a new position. A *Demonstrate* event occurs whenever a large number of people come together in a public area to protest or demand some sort of official action. *Demonstrate* events include, but are not limited to, protests, sit-ins, strikes, and riots. An example that includes two of such events is the phrase *Les groupes féministes organisent une manifestation en faveur du droit de vote pour les femmes (Feminist groups organise protest for women's rights to vote.)*. where *manifestation* triggers a *Demonstrate* and *vote* triggers an *Elect*.

The eight annotated French articles contain: 13 *Elect* events, six *Demonstrate*, and six *Meet* events. One could notice in Table 13 that the *Meet* events are generally detected due to, most probably, similarity of the trigger words in French and English, while the *Elect* events have a low recall. We assume that, because most of the events that were annotated as *Elect* were in the context of *droit de vote (right to vote*, these are probably not necessarily expressing elections, but rather a semantic concept.

7.2.2 Evaluation on a German NewsEye Subset: International Women's Day

The German articles range between 1911 and 1933 in order to analyse the events organised on or around the International Women's Day. For this subset, we chose several types of events presented in Table 14. The number of annotated events are presented in Figure 21. Regarding the amounts of events, one can notice that between 1914 and 1916, more events regarding *gatherings* or *movements* (these are revealed by the *Transport* and *Meet* event types), while several *attacks* increase between 1916 and 1933. To understand the meaning of the event types in a deeper analysis, we detail several types in the following paragraphs.

Event Type	Event Subtype
Conflict	Attack
Life	Death, Killing, Injure
Justice	Execution

Table 14: The event types and subtypes for the *International Women's Day* German subset.

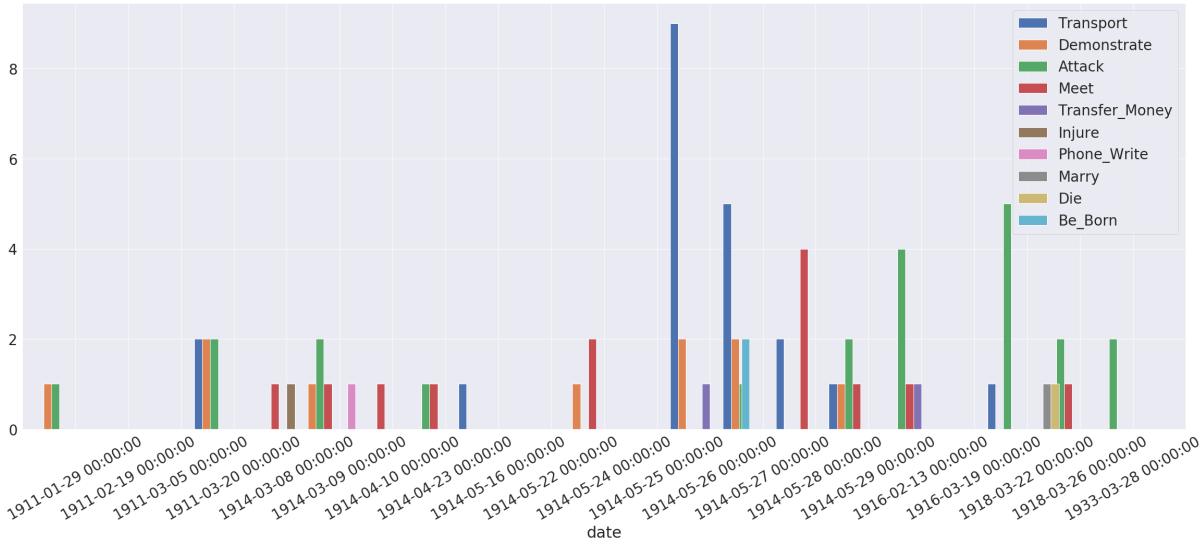

Figure 21: The number of annotated events in German.

An example from NewsEye German dataset of a *Transport* type of event is in the following phrase: *Zunächst, wurde der Wiener Frauenerwerbsverein besucht, wo die Damen von der Präsidentin Paula Thaienburg empfangen wurden.* (*First, the Vienna Women's Employment Association was visited, where the women were received by President Paula Thaienburg.*), where besucht means *visited* at it represents the event trigger for the *Transport* type of event.

Event Type	Event Subtype	P	R	F1
BERT-multilingual-uncased				
Conflict	Attack	33.33	4.55	8
All Be-Born	100	100	100	
	Demonstrate	50	9.09	15.38
	Die	0	0	0
	Injure	100	100	100
	Marry	100	100	100
	Meet	50	15.38	23.53
	Phone-Write	0	0	0
	Transfer-Money	0	0	0
	Transport	14.29	9.52	11.43
		35.71	13.33	19.42
BERT-multilingual-uncased +*Entity Type Markers*$^{+}$				
	Attack	27.27	42.86	33.33
	Be-Born	100	100	100
	Demonstrate	47.06	66.67	55.17
	Die	100	100	100
	Injure	50.0	100	66.67
	Marry	100	100	100
	Meet	83.33	38.46	52.63
	Phone-Write	50	100	66.67
	Transfer-Money	25.00	50.00	33.33
	Transport	41.67	23.81	30.30
		43.21	46.05	**44.59**

Table 15: Evaluation of NewsEye German event detection.

Demonstrate and *Attack* are subtypes of the *Conflict* event type. An *Attack* event is defined as a violent physical act causing harm or damage. For example, in *Um diesen ersehnten Zustand herbeizuführen, entsenden wir unseren Schwestern in der ganzen Welt unsere Grüße und rufen sie auf, beim internationalen Frauentag mit uns gemeinsam gegen die Fortdauer des Krieges zu demonstrieren.* (*In order to bring about this desired state, we send our greetings to our sisters all over the world and call on them to demonstrate together with us against the continuation of the war on International Women's Day.*), the triggers are: for *Demonstrate*, demonstrieren, and for *Attack*, Krieges. There are thus, in this case, two mentions of different types of events.

A total of 14 *Meet* events were annotated, along with 11 *Demonstrate* events. The other types are as follows: 22 *Attack*, two *Be-Born*, one *Die* event, one *Injure*, one *Marry*, one *Phone-Write*, two *Transfer-Money* events, and 21 *Transport*.

We can observe from Table 15 that in the results for the model that does not utilise entities, the performance drop significantly, while their presence and their preliminary detection proves to be efficient.

7.2.3 Evaluation on a French NewsEye Subset: Death Punishment Abolition

Since newspapers were always an important medium for the dissemination of public and political opinions, we created our dataset by selecting a subset of 2,655 French articles that mentioned the keyword "guillotine" (same meaning as in English) and "death penalty" ("peine de mort") published between 1900 and 1944 from the following newspapers: *Le Matin*, *L'Œuvre* and *Le Gaulois*.

A recent impacful work presented an effort to gather requirements about the linguistic annotation of events in historical texts from domain experts [7]. This research suggested that a careful adaptation of existing annotation schemes is necessary to meet the requirements of experts in the domain. Thus, we defined an event to be consistent with ACE 2005 [112] and chose the event types and subtypes according to their annotation guidelines[27]. We decided to approach three different event types: conflictual events (*Conflict*), life-related (*Life*), and criminal justice events (*Justice*), with a set of event subtypes as presented in Table 16 and 17.

Event Type	Event Subtype
Conflict	Attack
Life	Death, Killing, Injure
Justice	Execution

Table 16: The event types and subtypes mapped to FrameNet.

We then automatically assigned a frame category to each event type by consulting the English FrameNet database[28]. Next, we associated, for each subtype, the FrameNet lexical units which are the words that evoke each frame and can be viewed as event triggers.

Event Type	Event Subtype
Conflict	Attack {Assailant, Victim}, Explosion {Place, Victim}, Destroying {Destroyer, Patient}, Protest {Protester, Place}
Life	Death {Protagonist, Place}, Killing {Killer, Victim}, Injure {Agent, Victim}, Dead or alive {Protagonist}
Justice	Imprisonment {Authorities, Prisoner}, Execution {Executed, Executioner}

Table 17: The event types and subtypes $\{Arguments\}$ mapped to FrameNet.

For a preliminary assessment of the correctness of our approach, we manually annotated 207 sentences randomly chosen from the dataset, among the sentences that contain at least one lexical unit. The results, summarized in Table 18, reveal the capacity of our approach for extracting events, while establishing a strong baseline. However, we notice that the scores are rather imbalanced, favoring recall, which could indicate a close similarity between the chosen event types[29].

For entity recognition, we used a recently proposed model for fine-grained named entity recognition in historical documents [12, 111, 113] that consists of a hierarchical architecture that includes a stack of

[27] https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf

[28] An example of the *Justice.Execution* frame can be viewed at https://framenet2.icsi.berkeley.edu/fnReports/data/frameIndex.xml?frame=Execution.

[29] We observed, for example, that "guillotiner" (*to guillotine*) was occasionally confused with an event of type Life.Killing. However, it can also represent an event of type Justice.Execute.

Event Type	Event Subtype	P	R	F1
BERT-multilingual-uncased				
Conflict	Attack	13.31	18.22	15.41
Life	Die	42.30	19.60	26.83
Justice	Execute	40.00	10.00	16.00
		31.87	15.94	19.40
BERT-multilingual-uncased+*Entity Type Markers*				
Conflict	Attack	20.10	18.21	19.21
Life	Die	30.82	21.41	25.30
Justice	Execute	100.0	1.50	30.0
		50.30	13.70	24.83
FrameNet-based Method				
Conflict	Attack	76.92	90.91	83.33
	Injure	60.40	92.42	73.05
Life	Death	74.07	83.33	78.43
	Killing	45.16	90.32	60.22
Justice	Execute	100.00	45.00	62.07
		60.38	84.21	70.33

Table 18: Evaluation scores for trigger event detection. F1-micro

Transformer layers [2] on top of a BERT encoder. For our study, we detected the persons (PER)[30], and we selected the most discussed (> 15 mentions) and the least discussed person entities ($= 1$)[31]. By looking for entities that suddenly spike in popularity on a given date, we can identify the frequency domain that corresponds to a trending event. We map these entities with the identified subjects and objects in order to have a view regarding the event participants.

This period covers the later years of the Third Republic, the World War I and II, and the interwar period, when newspapers had changed different political views. Due to the digitization and article separation processes, some of the articles contained an insignificant amount of tokens, and thus, we removed the articles with less than ten tokens[32].

In the 1900s, in France, a general debate took place regularly on the question of the abolition of the capital punishment. Therefore, in 1906, the parliamentarians called for another debate. Figure 22 shows a ramp-up trend pattern during this year. This is explained by the fact that Armand Fallières (1841-1931), a convinced abolitionist, was elected President of the Republic and immediately put the debate on the agenda, decision supported by politicians such as Guyot-Dessaigne and Briand (solid orange and red lines). Unfortunately, before the vote, a terrible crime occurred (news item exploited to the maximum by the press, the spike being visible in solid green in Figure 22), when in Paris, 1907, a child[33] was killed by a family friend, Albert Soleilland. However, he did not get the death penalty, which produces the highest spike (green solid line). After a three-year hiatus, executions resumed between 1909 and 1929. During this period of time, we can observe frequent detected events (Life.Cause_harm

[30] Future work will include a larger set of entity types.

[31] Previous results on the collection XXX, with the best performing system BERT-1×Transformer-CRF for French and CamemBERT-large as the encoder, resulted in a precision of 75%, a recall of 70.6% and an F1-micro score of 72.6%.

[32] This threshold was chosen after we manually checked the dismissed articles, in order to verify if any relevant document was being removed.

[33] The name was not disclosed in the press.

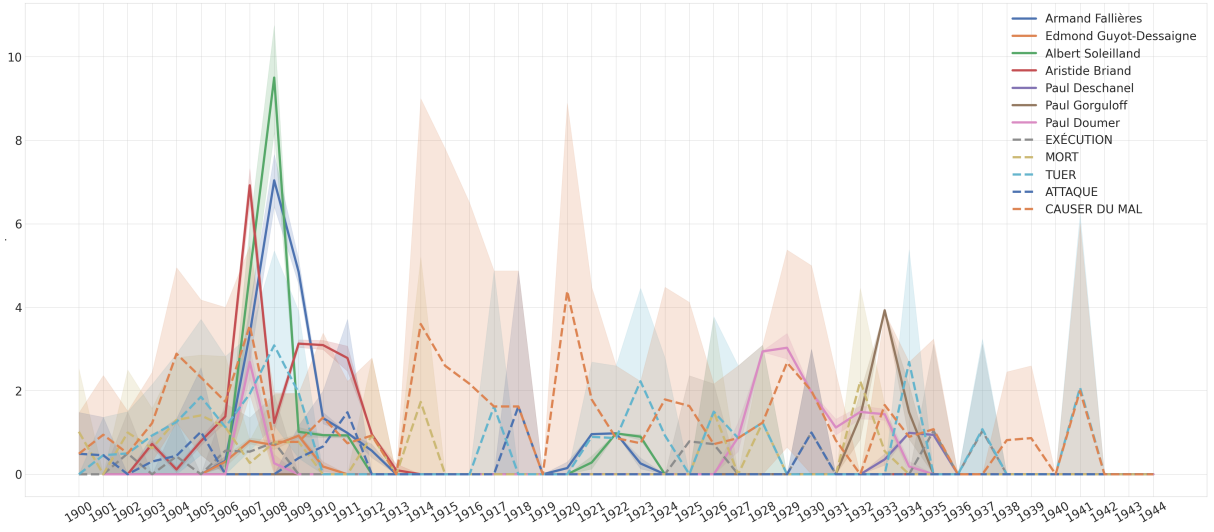

Figure 22: Extracted events and entities in French newspapers 1900-1944: frequency (thick or dotted lines) and standard deviations (shaded areas).

(dotted green), Conflict.Killing (dotted blue), and Justice.Execution (dotted brown)).

Two other interesting trends can be spotted in the periods 1931-1935 and 1939-1942. During the first one, Paul Doumer (president of France in 1931–1932) was assassinated by Paul Gorgulov, a Russian emigrant who later was executed by guillotine (solid pink and brown). The second refers to a spike in discussions about the last person publicly executed by guillotine in France (1939, Execution (*Exécution*), brown dotted line).

8 Conclusions

In this final deliverable, we approached the task of event detection, using different systems and different datasets, we studied the impact of the level of degradation of images on the performance of the systems, and finally, we evaluated several methods on a sample of annotated NewsEye documents. We chose two datasets, one was created for the DAnIEL system (Data Analysis for Information Extraction in any Language) and the other one was the ACE2005 corpora provided by the ACE evaluation[34]. The DAnIEL dataset consists of a large multilingual number of collected documents focused on a single type of event, epidemic event. ACE2005 dataset covers the most common types of events (attacks, births, deaths, meetings, demonstrations) from national and international news (from a variety of sources selected from broadcast news programs, newspapers, news wire reports, internet sources or transcribed audio).

We chose two baseline models: one based on the DAnIEL system which exploits the global structure of news regarding only epidemic outbreaks and a neural-based approach that consisted in a convolutional neural network (CNN) applied to a local context around potential event triggers, independent of the type of data. We experimented on how well the models perform in perfect conditions and also, with added noise from aging documents, scanning and ATR process that can affect the quality of these event detection systems, with regard to the chosen datasets. We conclude that, in these two cases, the event detection is prone to errors from OCR, depending on the level of data imbalance. The DAnIEL dataset was highly imbalanced and the variability in results was lower than in the case of the ACE 2005,

[34] https://catalog.ldc.upenn.edu/ldc2006t06

and thus the probability of the small amount of annotated words as events to be affected becomes quite low. Moreover, ACE 2005 has a much higher number of events and event types in almost every document, 92.32% of the documents are relevant, while in DAnIEL dataset, only 10.14%. Comparing the models, the CNN-based model is more impacted by the effects of the noise added to the images than the DAnIEL system, and we hold responsible the word-level representations. The lesser impact on the DAnIEL system, meanwhile, can be also motivated by the fact that the model uses external resources in order to predict the presence of an event. One disadvantage in future use of this model might be its exclusive applicability in news focused on epidemic events, and the amount of effort in order to adapt it to other domains (e.g., Wikipedia seeds for different domains need to be provided). An advantage that is common to both models is their language independence.

Further, we developed complex architectures for event detection based on fine-tuning large language models, with different paradigms (sequential data classification and question answering) and also taking advantage of the presence of entities, in order to tackle even more the multilingual characteristic of NewsEye data and the errors that are being propagated from the digitisation process. We also proposed un unsupervised technique based on available resources (FrameNet) that could help in alleviating the need for manual annotation by also providing a practical course of action towards unsupervised event extraction from multilingual digitized and historical documents. We, thus, experiment with two annotated NewsEye datasets, for French and for German. In order to adapt our models to the NewsEye languages, we utilised pre-trained multilingual models. Due to the lack of annotated data, we adopted a zero-shot technique by training our models on the ACE2005 dataset, dataset that contains a more detailed and fine-grained set of event types, and predicting on the datasets in the NewsEye languages. The experiments proved not only that our proposed models are able to detect events even though they were never seen in the specified language, but also with an impressive precision. We consider that a further introspection of this type of technique for event detection could reveal interesting depths regarding the analysis of historical documents.

References

[1] Gaël Lejeune, Romain Brixtel, Antoine Doucet, and Nadine Lucas. “Multilingual Event Extraction for Epidemic Detection”. In: *Artificial intelligence in medicine* 65 (July 2015). DOI: 10.1016/j.artmed.2015.06.005.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. “Attention is all you need”. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).

[4] Jerry R. Hobbs and Moore Robert C. *Formal Theories of the Commons. World*. Ablex Publishing Corporation, 1998.

[5] Ralph Grishman and Beth Sundheim. “Message understanding conference-6: A brief history”. In: *COLING 1996*. 1996, pp. 466–471.

[6] George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. “The Automatic Content Extraction (ACE) Program – Tasks, Data, and Evaluation”. In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC’04)*. Lisbon, Portugal: European Language Resources Association (ELRA), May 2004. URL: http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf.

[7] Rachele Sprugnoli. “Event Detection and Classification for the Digital Humanities”. PhD thesis. University of Trento, 2018.

[8] Nancy Chinchor, David D Lewis, and Lynette Hirschman. “Evaluating message understanding systems: an analysis of the third message understanding conference (MUC-3)”. In: *Computational linguistics* 19.3 (1993), pp. 409–449.

[9] Jacqueline Aguilar, Charley Beller, Paul McNamee, Benjamin Van Durme, Stephanie Strassel, Zhiyi Song, and Joe Ellis. “A comparison of the events and relations across ace, ere, tac-kbp, and framenet annotation standards”. In: *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*. 2014, pp. 45–53.

[10] Gaël Lejeune, Antoine Doucet, Roman Yangarber, and Nadine Lucas. “Filtering news for epidemic surveillance: towards processing more languages with fewer resources”. In: *Proceedings of the 4th Workshop on Cross Lingual Information Access*. 2010, pp. 3–10.

[11] David Miller, Sean Boisen, Richard Schwartz, Rebecca Stone, and Ralph Weischedel. “Named entity extraction from noisy input: speech and OCR”. In: *Proceedings of the sixth conference on Applied natural language processing*. Seattle, Washington, USA: Association for Computational Linguistics, 2000, pp. 316–324.

[12] Emanuela Boroş, Ahmed Hamdi, Elvys Linhares Pontes, Luis-Adrián Cabrera-Diego, José G Moreno, Nicolas Sidere, and Antoine Doucet. “Alleviating Digitization Errors in Named Entity Recognition for Historical Documents”. In: *Proceedings of the 24th Conference on Computational Natural Language Learning*. 2020, pp. 431–441.

[13] Kepa Joseba Rodriquez, Mike Bryant, Tobias Blanke, and Magdalena Luszczynska. “Comparison of named entity recognition tools for raw OCR text”. In: *11th Conference on Natural Language Processing, KONVENS 2012, Empirical Methods in Natural Language Processing, September 19-21, 2012*. Ed. by Jeremy Jancsary. Vol. 5. Scientific series of the ÖGAI. Vienna: ÖGAI, Wien, Österreich, 2012, pp. 410–414. URL: http://www.oegai.at/konvens2012/proceedings/60_rodriquez12w/.

[14] Ahmed Hamdi, Axel Jean-Caurant, Nicolas Sidere, Mickaël Coustaty, and Antoine Doucet. “An analysis of the performance of named entity recognition over OCRed documents”. In: *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. Illinois, USA: IEEE, 2019, pp. 333–334.

[15] Daniel van Strien, Kaspar Beelen, Mariona Coll Ardanuy, Kasra Hosseini, Barbara McGillivray, and Giovanni Colavizza. “Assessing the Impact of OCR Quality on Downstream NLP Tasks”. In: *ICAART 2020 - Proceedings of the 12th International Conference on Agents and Artificial Intelligence* 1 (2020), pp. 484–496.

[16] Stephen Mutuvi, Antoine Doucet, Moses Odeo, and Adam Jatowt. “Evaluating the impact of OCR errors on topic modeling”. In: *International Conference on Asian Digital Libraries*. Berlin, Germany: Springer, 2018, pp. 3–14.

[17] Emanuela Boros, Nhu Khoa Nguyen, Gaël Lejeune, and Antoine Doucet. “Assessing the Impact of OCR Noise on Multilingual Event Detection over Digitised Documents”. In: *International Journal on Digital Libraries* (2022).

[18] Nhu Khoa Nguyen, Emanuela Boros, Gaël Lejeune, and Antoine Doucet. “Impact Analysis of Document Digitization on Event Extraction”. In: *Proceedings of the 4th Workshop on Natural Language for Artificial Intelligence (NL4AI 2020) co-located with the 19th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2020), Online, November 25-27, 2020*. Vol. 2735. CEUR Workshop Proceedings. 2020, pp. 17–28.

[19] George Krupka, Paul Jacobs, Lisa Rau, and Lucja Iwańska. “GE: Description of the NLToolset System as Used for MUC-3”. In: *3rd Conference on Message understanding*. 1991, pp. 144–149.

[20] Jerry R Hobbs, Douglas Appelt, Mabry Tyson, John Bear, and David Israel. “SRI International: Description of the FASTUS system used for MUC-4”. In: *4th Conference on Message understanding*. 1992, pp. 268–275.

[21] Ellen Riloff. “Automatically generating extraction patterns from untagged text”. In: *AAAI'96*. 1996, pp. 1044–1049.

[22] Ellen Riloff. “An empirical study of automated dictionary construction for information extraction in three domains”. In: *Artificial intelligence* 85.1 (1996), pp. 101–134.

[23] Roman Yangarber, Ralph Grishman, Pasi Tapanainen, and Silja Huttunen. “Automatic acquisition of domain knowledge for information extraction”. In: *18th Internation Conference on Computational Linguistics (COLING 2000)*. 2000, pp. 940–946.

[24] Dayne Freitag. “Information extraction from HTML: Application of a general machine learning approach”. In: *AAAI'98*. 1998, pp. 517–523.

[25] Hai Leong Chieu, Hwee Tou Ng, and Yoong Keok Lee. “Closing the gap: Learning-based information extraction rivaling knowledge-engineering methods”. In: *41st international Annual Meeting on Association for Computational Linguistics (ACL-2003)*. 2003, pp. 216–223.

[26] Mihai Surdeanu, Jordi Turmo, and Alicia Ageno. “A hybrid approach for the acquisition of information extraction patterns”. In: *EACL-2006 Workshop on Adaptive Text Extraction and Mining (ATEM 2006)*. 2006, pp. 48–55.

[27] Heng Ji and Ralph Grishman. “Refining Event Extraction through Cross-Document Inference”. In: *Proceedings of ACL-08: HLT*. Columbus, Ohio: Association for Computational Linguistics, June 2008, pp. 254–262. URL: https://www.aclweb.org/anthology/P08-1030.

[28] Siddharth Patwardhan and Ellen Riloff. “A unified model of phrasal and sentential evidence for information extraction”. In: *2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*. 2009, pp. 151–160.

[29] Shasha Liao and Ralph Grishman. “Using Document Level Cross-Event Inference to Improve Event Extraction”. In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden: Association for Computational Linguistics, July 2010, pp. 789–797. URL: https://www.aclweb.org/anthology/P10-1081.

[30] Ruihong Huang and Ellen Riloff. “Peeling Back the Layers: Detecting Event Role Fillers in Secondary Contexts”. In: *ACL 2011*. 2011, pp. 1137–1147.

[31] Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. “Using cross-entity inference to improve event extraction”. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics. 2011, pp. 1127–1136.

[32] Qi Li, Heng Ji, and Liang Huang. “Joint Event Extraction via Structured Prediction with Global Features”. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics, Aug. 2013, pp. 73–82. URL: https://www.aclweb.org/anthology/P13-1008.

[33] Ofer Bronstein, Ido Dagan, Qi Li, Heng Ji, and Anette Frank. “Seed-Based Event Trigger Labeling: How far can event descriptions get us?” In: *ACL (2)*. 2015, pp. 372–376.

[34] Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. “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*. Vol. 1. 2015, pp. 167–176.

[35] Thien Huu Nguyen and Ralph Grishman. “Event Detection and Domain Adaptation with 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 2: Short Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 365–371. DOI: 10.3115/v1/P15-2060. URL: https://www.aclweb.org/anthology/P15-2060.

[36] Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. “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*. 2016, pp. 300–309.

[37] Xiaocheng Feng, Lifu Huang, Duyu Tang, Heng Ji, Bing Qin, and Ting Liu. “A language-independent neural network for event detection”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Vol. 2. 2016, pp. 66–71.

[38] Emanuela Boros. “Neural Methods for Event Extraction”. PhD thesis. Université Paris Sud, 2018.

[39] Shulin Liu, Yubo Chen, Shizhu He, Kang Liu, Jun Zhao, et al. “Leveraging framenet to improve automatic event detection”. In: (2016).

[40] Wei Li, Dezhi Cheng, Lei He, Yuanzhuo Wang, and Xiaolong Jin. “Joint event extraction based on hierarchical event schemas from framenet”. In: *IEEE Access* 7 (2019), pp. 25001–25015.

[41] Shulin Liu, Yubo Chen, Kang Liu, and Jun Zhao. “Exploiting Argument Information to Improve Event Detection via Supervised Attention Mechanisms”. In: *55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*. Vancouver, Canada, 2017, pp. 1789–1798.

[42] Collin F Baker, Charles J Fillmore, and John B Lowe. “The berkeley framenet project”. In: *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*. 1998, pp. 86–90.

[43] Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. “Exploring Pre-trained Language Models for Event Extraction and Generation”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019, pp. 5284–5294.

[44] Tongtao Zhang, Heng Ji, and Avirup Sil. “Joint entity and event extraction with generative adversarial imitation learning”. In: *Data Intelligence* 1.2 (2019), pp. 99–120.

[45] Yu Hong, Wenxuan Zhou, Jingli Zhang, Guodong Zhou, and Qiaoming Zhu. “Self-regulation: Employing a generative adversarial network to improve event detection”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2018, pp. 515–526.

[46] Xinya Du and Claire Cardie. “Event Extraction by Answering (Almost) Natural Questions”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 671–683. DOI: 10.18653/v1/2020.emnlp-main.49. URL: https://aclanthology.org/2020.emnlp-main.49.

[47] Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu. “Event extraction as machine reading comprehension”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020, pp. 1641–1651.

[48] Ruihong Huang and Ellen Riloff. “Bootstrapped training of event extraction classifiers”. In: *13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012)*. 2012, pp. 286–295.

[49] Kiyoshi Sudo, Satoshi Sekine, and Ralph Grishman. “An improved extraction pattern representation model for automatic IE pattern acquisition”. In: *41st Annual Meeting on Association for Computational Linguistics (ACL-03)*. 2003, pp. 224–231.

[50] Mark Stevenson and Mark A Greenwood. “A semantic approach to IE pattern induction”. In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 2005, pp. 379–386.

[51] Siddharth Patwardhan and Ellen Riloff. “Effective information extraction with semantic affinity patterns and relevant regions”. In: *2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*. 2007, pp. 717–727.

[52] Siddharth Patwardhan. “Widening the field of view of information extraction through sentential event recognition”. PhD thesis. University of Utah, 2010.

[53] Ralph Grishman, David Westbrook, and Adam Meyers. “NYU’s English ACE 2005 system description”. In: *ACE* 5 (2005).

[54] David Ahn. “The stages of event extraction”. In: *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*. Association for Computational Linguistics. 2006, pp. 1–8.

[55] Peifeng Li, Qiaoming Zhu, and Guodong Zhou. “Argument Inference from Relevant Event Mentions in Chinese Argument Extraction.” In: *ACL (1)*. 2013, pp. 1477–1487.

[56] Rahul Gupta and Sunita Sarawagi. “Domain adaptation of information extraction models”. In: *ACM SIGMOD Record* 37.4 (2009), pp. 35–40.

[57] Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. “A hierarchical neural autoencoder for paragraphs and documents”. In: *arXiv preprint arXiv:1506.01057* (2015).

[58] Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. “Class-based n-gram models of natural language”. In: 18 (1992), pp. 467–479.

[59] George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. “Introduction to WordNet: An on-line lexical database”. In: *International journal of lexicography* 3.4 (1990), pp. 235–244.

[60] Shulin Liu, Kang Liu, Shizhu He, and Jun Zhao. “A Probabilistic Soft Logic Based Approach to Exploiting Latent and Global Information in Event Classification”. In: *Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*. Phoenix, Arizona, 2016, pp. 2993–2999.

[61] Thien Huu Nguyen and Ralph Grishman. “Modeling Skip-Grams for Event Detection with Convolutional Neural Networks”. In: *Proceedings of EMNLP*. 2016.

[62] Abhyuday N Jagannatha and Hong Yu. “Bidirectional RNN for Medical Event Detection in Electronic Health Records”. In: *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Vol. 2016. NIH Public Access. 2016, pp. 473–482.

[63] Thien Huu Nguyen and Ralph Grishman. “Graph Convolutional Networks With Argument-Aware Pooling for Event Detection”. In: *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*. New Orleans, Louisiana, USA, 2018.

[64] Shaoyang Duan, Ruifang He, and Wenli Zhao. “Exploiting Document Level Information to Improve Event Detection via Recurrent Neural Networks”. In: *Eighth International Joint Conference on Natural Language Processing (IJCNLP 2017)*. Taipei, Taiwan: Asian Federation of Natural Language Processing, 2017, pp. 352–361.

[65] Yue Zhao, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. “Document Embedding Enhanced Event Detection with Hierarchical and Supervised Attention”. In: *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 414–419. URL: http://aclweb.org/anthology/P18-2066.

[66] Yu Hong, Wenxuan Zhou, jingli, Guodong Zhou, and Qiaoming Zhu. “Self-regulation: Employing a Generative Adversarial Network to Improve Event Detection”. In: *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 515–526. URL: http://aclweb.org/anthology/P18-1048.

[67] Jian Liu, Yubo Chen, Kang Liu, and Jun Zhao. “Event Detection via Gated Multilingual Attention Mechanism”. In: *Thirty-second AAAI Conference on Artificial Intelligence (AAAI-18)*. New Orleans, Louisiana, 2018.

[68] Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. “Adversarial training for weakly supervised event detection”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, pp. 998–1008.

[69] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. “Entity, relation, and event extraction with contextualized span representations”. In: *arXiv preprint arXiv:1909.03546* (2019).

[70] Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. “Distilling Discrimination and Generalization Knowledge for Event Detection via Delta-Representation Learning”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019, pp. 4366–4376.

[71] Xiao Liu, Zhunchen Luo, and Heyan Huang. “Jointly multiple events extraction via attention-based graph information aggregation”. In: *arXiv preprint arXiv:1809.09078* (2018).

[72] Shulin Liu, Yubo Chen, Kang Liu, Jun Zhao, et al. “Exploiting argument information to improve event detection via supervised attention mechanisms”. In: (2017).

[73] Emanuela Boros, Jose G. Moreno, and Antoine Doucet. “Exploring Entities in Event Detection as Question Answering”. In: *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, April 10-14, 2022, Trondheim, Norway, Proceedings*. Lecture Notes in Computer Science. Springer, 2022.

[74] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. “Squad: 100,000+ questions for machine comprehension of text”. In: *arXiv preprint arXiv:1606.05250* (2016).

[75] Pranav Rajpurkar, Robin Jia, and Percy Liang. “Know what you don’t know: Unanswerable questions for SQuAD”. In: *arXiv preprint arXiv:1806.03822* (2018).

[76] Ryan Benjamin Shaw. *Events and periods as concepts for organizing historical knowledge*. University of California, Berkeley, 2010.

[77] Sarah Oberbichler, Emanuela Boroş, Antoine Doucet, Jani Marjanen, Eva Pfanzelter, Juha Rautiainen, Hannu Toivonen, and Mikko Tolonen. “Integrated interdisciplinary workflows for research on historical newspapers: Perspectives from humanities scholars, computer scientists, and librarians”. In: *Journal of the Association for Information Science and Technology* 73.2 (2022), pp. 225–239.

[78] Nancy Ide and David Woolner. “Exploiting Semantic Web Technologies for Intelligent Access to Historical Documents”. In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. Lisbon, Portugal: European Language Resources Association (ELRA), May 2004. URL: http://www.lrec-conf.org/proceedings/lrec2004/pdf/248.pdf.

[79] Federico Boschetti, Andrea Cimino, Felice Dell'Orletta, Gianluca Lebani, Lucia Passaro, Paolo Picchi, Giulia Venturi, Simonetta Montemagni, and Alessandro Lenci. “Computational analysis of historical documents: An application to Italian war bulletins in World War I and II”. In: *Workshop on Language resources and technologies for processing and linking historical documents and archives (LRT4HDA 2014)*. ELRA. 2014, pp. 70–75.

[80] Agata Cybulska and Piek Vossen. “Historical event extraction from text”. In: *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. 2011, pp. 39–43.

[81] Agata Cybulska and Piek Vossen. “Event Models for Historical Perspectives: Determining Relations between High and Low Level Events in Text, Based on the Classification of Time, Location and Participants.” In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA), May 2010. URL: http://www.lrec-conf.org/proceedings/lrec2010/pdf/205_Paper.pdf.

[82] Elizabeth Boschee, Premkumar Natarajan, and Ralph Weischedel. “Automatic extraction of events from open source text for predictive forecasting”. In: *Handbook of Computational Approaches to Counterterrorism*. Springer, 2013, pp. 51–67.

[83] Gaël Lejeune, Romain Brixtel, Charlotte Lecluze, Antoine Doucet, and Nadine Lucas. “Added-Value of Automatic Multilingual Text Analysis for Epidemic Surveillance”. In: *Artificial Intelligence in Medicine - 14th Conference on Artificial Intelligence in Medicine, AIME 2013, Murcia, Spain, May 29 - June 1, 2013. Proceedings*. Vol. 7885. Lecture Notes in Computer Science. Springer, 2013, pp. 284–294.

[84] Nadine Lucas. “Modélisation différentielle du texte, de la linguistique aux algorithmes”. PhD thesis. Université de Caen, 2009.

[85] Felix Hamborg, Soeren Lachnit, Moritz Schubotz, Thomas Hepp, and Bela Gipp. “Giveme5W: Main Event Retrieval from News Articles by Extraction of the Five Journalistic W Questions”. In: *Transforming Digital Worlds*. Cham: Springer International Publishing, Mar. 2018, pp. 356–366. DOI: 10.1007/978-3-319-78105-1_39.

[86] Emmanuel Giguet and Nadine Lucas. “La détection automatique des citations et des locuteurs dans les textes informatifs”. In: *Le discours rapporté dans tous ses états: Question de frontières* (2004), pp. 410–418.

[87] Esko Ukkonen. “Maximal and minimal representations of gapped and non-gapped motifs of a string”. In: *Theoretical Computer Science* 410 (Oct. 2009), pp. 4341–4349. DOI: 10.1016/j.tcs.2009.07.015.

[88] Yoon Kim. “Convolutional neural networks for sentence classification”. In: *arXiv preprint arXiv:1408.5882* (2014).

[89] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient Estimation of Word Representations in Vector Space”. In: *1st International Conference on Learning Representations, ICLR 2013*. Ed. by Yoshua Bengio and Yann LeCun. Scottsdale, Arizona, USA: IEEE, 2013, p.

[90] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. “Relation Classification via Convolutional Deep Neural Network.” In: *COLING*. 2014, pp. 2335–2344.

[91] Cıcero Nogueira Dos Santos and Maira Gatti. “Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts.” In: *COLING*. 2014, pp. 69–78.

[92] Chenxi Zhu, Xipeng Qiu, Xinchi Chen, and Xuanjing Huang. “A re-ranking model for dependency parser with recursive convolutional neural network”. In: *arXiv preprint arXiv:1505.05667* (2015).

[93] Samuel L. Smith, Pieter-Jan Kindermans, and Quoc V. Le. “Don’t Decay the Learning Rate, Increase the Batch Size”. In: *CoRR* abs/1711.00489 (2017), p. arXiv: 1711.00489. URL: http://arxiv.org/abs/1711.00489.

[94] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. “Convolutional Sequence to Sequence Learning”. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML’17. Sydney, NSW, Australia: JMLR.org, 2017, pp. 1243–1252.

[95] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. “Neural architectures for named entity recognition”. In: *arXiv preprint arXiv:1603.01360* (2016).

[96] Xuezhe Ma and Eduard Hovy. “End-to-end sequence labeling via bi-directional lstm-cnns-crf”. In: *arXiv preprint arXiv:1603.01354* (2016).

[97] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. “Matching the blanks: Distributional similarity for relation learning”. In: *arXiv preprint arXiv:1906.03158* (2019).

[98] Emanuela Boros, Jose G. Moreno, and Antoine Doucet. “Event Detection with Entity Markers”. In: *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part II*. Vol. 12657. Lecture Notes in Computer Science. Springer, 2021, pp. 233–240.

[99] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. “Google’s neural machine translation system: Bridging the gap between human and machine translation”. In: *arXiv preprint arXiv:1609.08144* (2016).

[100] Matthew Honnibal and Ines Montani. “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing”. In: *Unpublished software application* (2017). URL: https://spacy.io.

[101] Nicholas Journet, Muriel Visani, Boris Mansencal, Kieu Van-Cuong, and Antoine Billy. “DocCreator: A New Software for Creating Synthetic Ground-Truthed Document Images”. In: *Journal of imaging* 3.4 (2017), p. 62.

[102] Ray Smith. “An overview of the Tesseract OCR engine”. In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. Vol. 2. IEEE. USA: IEEE Computer Society, 2007, pp. 629–633.

[103] Lutz Prechelt. “Early stopping-but when?” In: *Neural Networks: Tricks of the trade* (1998), pp. 553–553.

[104] Matthew D Zeiler. “ADADELTA: an adaptive learning rate method”. In: *arXiv preprint arXiv:1212.5701* (2012).

[105] Peilu Wang, Ruihua Sun, Hai Zhao, and Kai Yu. “A New Word Language Model Evaluation Metric for Character Based Languages”. In: *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*. Ed. by Maosong Sun, Min Zhang, Dekang Lin, and Haifeng Wang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 315–324.

[106] Gaël Lejeune and Lichao Zhu. “A New Proposal for Evaluating Web Page Cleaning Tools”. In: *Computacion y Sistemas* 22.4 (2018), pp. 1249–1258.

[107] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186.

[108] Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. “On identifiability in transformers”. In: *International Conference on Learning Representations*. 2019.

[109] Andreas Madsen. “Visualizing memorization in RNNs”. In: *Distill* (2019). DOI: 10.23915/distill.00016.

[110] Axel Jean-Caurant and Antoine Doucet. “Accessing and Investigating Large Collections of Historical Newspapers with the NewsEye Platform”. In: *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*. 2020, pp. 531–532.

[111] Emanuela Boros, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Ahmed Hamdi, Jose G. Moreno, Nicolas Sidère, and Antoine Doucet. “Robust Named Entity Recognition and Linking on Historical Multilingual Documents”. In: *CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*. Ed. by Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol. CEUR-WS, 2020.

[112] Christopher Walker, Strassel Stephanie, Medero Julie, and Maeda Kazuaki. “ACE 2005 multilingual training corpus”. In: *Technical report, Linguistic Data Consortium* (2005).

[113] Emanuela Boros, Ahmed Hamdi, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Jose Moreno, Nicolas Sidere, and Antoine Doucet. “Atténuer les erreurs de numérisation dans la reconnaissance d’entités nommées pour les documents historiques”. In: *COnférence en Recherche d’Informations et Applications-CORIA 2021, French Information Retrieval Conference,* 2021.