

Article

An Event Extraction Approach Based on a Multi-Round Q&A Framework

Li He ^{1,2}, Xiya Zhao ^{1,2}, Liang Zhao ^{3,4,*} and Qing Zhang ¹¹ School of Information Science and Technology, North China University of Technology, Beijing 100144, China² CNONIX National Standard Application and Promotion Lab, Beijing 100144, China³ School of Information Management, Wuhan University, Wuhan 430072, China⁴ Key Laboratory of Semantic Publishing and Knowledge Service of the National Press and Publication Administration, Wuhan University, Wuhan 430072, China

* Correspondence: liangzhao@whu.edu.cn

Abstract: Event extraction aims to present unstructured text containing event information in a structured form to help people quickly mine the target information. Most of the traditional event extraction methods focus on the design of complex neural network models, which rely on a large amount of annotated data to train the models. In recent years, some researchers have proposed the use of machine reading comprehension models for event extraction; however, the existing methods are limited to the single-round question-and-answer model, ignoring the dependency relation between the elements of event arguments. In addition, the existing methods do not fully utilize knowledge such as a priori information. To address these shortcomings, a multi-round Q&A framework is proposed for event extraction, which extends the existing methods in two aspects: first, by constructing a multi-round extraction problem framework, the model can effectively exploit the hierarchical dependencies among the argument elements; second, the question-and-answer framework is populated with historical answer information encoding slots, which are integrated into the multi-round Q&A process to assist in inference. Finally, experimental results on a publicly available dataset show that the proposed model achieves superior results compared to existing methods.



Citation: He, L.; Zhao, X.; Zhao, L.; Qing, Z. An Event Extraction Approach Based on a Multi-Round Q&A Framework. *Appl. Sci.* **2023**, *13*, 6308. <https://doi.org/10.3390/app13106308>

Academic Editor: Arcangelo Castiglione

Received: 10 April 2023

Revised: 14 May 2023

Accepted: 16 May 2023

Published: 22 May 2023



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

Keywords: event extraction; multi-round Q&A framework; historical answer information; machine reading comprehension

1. Introduction

With the constant influx of information on the Internet, all industries have produced a huge amount of information data. This information is characterized by diversity and disorder, and the vast majority of it is redundant, invalid, and worthless. On the contrary, valuable data is scarce, which has created some inevitable information overload problems. How can valuable information be extracted from massive and chaotic data by technical means, and how to classify, extract and reconstruct it so that people can easily and comprehensively obtain information from the exploding massive data is a problem that needs to be solved urgently.

Event extraction, i.e., identifying and extracting the corresponding event type and event element information from the text describing the event information, transforms the unstructured text into a structured event representation. In general, event extraction can be divided into two parts, i.e., event type detection and event element extraction. Among them, the main goal of event type detection is to identify the corresponding trigger words and determine the event type to which they belong, while the goal of event element extraction is to try to identify the attribute elements corresponding to the event type in the description text based on the event type identified in the previous step. For example, take the following example sentence.

Example 1. *British anti-terror police arrested Sharif in London on May 2.*

trigger word

First, in the event type detection task, we determine “arrested” as the trigger word in the sentence and determine the event type as Justice and Arrest-Jail, according to the trigger word. According to the event type, the corresponding event elements “British anti-terror police”, “Sharif”, “London”, and “May 2” were extracted and their corresponding roles were identified as Agent, Person, Time, and Place.

word vector tools

In research work on event extraction tasks, early approaches were based on pattern matching, which required significant human resources and had poor portability and low recall when migrating to new domain data because of its dependence on the specific form of the text. Therefore machine-learning-based methods gradually replace the traditional pattern matching methods. More typical machine learning methods include maximum entropy models, conditional random fields, support vector machines, etc. The core of these methods lies in obtaining feature representations based on datasets and models and classifying tasks into classification problems for processing. However, the performance of these methods is greatly affected by the features and can cause the problem of error accumulation. In recent years, most of the event extraction studies have been based on deep learning methods and progress with the evolution of deep learning models. Compared with the earlier methods, neural network models solve the problem of difficulty improving the learning ability of traditional machine learning methods and can also use the chain network structure of long short-term memory neural networks to model the contextual relations, which does not depend on external resources and further enhances the performance. After the prevalence of word vector tools, innovative cross-domain research results have been introduced to become the mainstream event extraction methods, such as the JMEE [1] model based on graph convolutional networks to represent sentence dependencies as graphs to obtain deeper semantic features, the Joint3EE [2] model that combines named entity recognition with trigger word and argument recognition tasks, and the MTL-CRF [3] model that treats event extraction as a sequence labeling task based on conditional random fields. However, such methods require a large amount of labeled data to train the model for the extraction task, which not only requires expensive labeled sample cost but also limits the diversity of events covered, and their reliance on upstream tasks inevitably generates the problem of error accumulation.

priori information to be utilized

In response to the above problems, some researchers have proposed the use of machine reading comprehension model for event extraction in the last two years. Such methods transform the event extraction task into a reading comprehension problem, which can take advantage of the pre-trained models on the one hand, and introduce a priori information into the problem on the other hand, further alleviating the dependence on the upstream task and the problem of incomplete retention of semantic information.

machine reading
comprehension model

multi round QA model

Although the approaches based on the machine reading comprehension model effectively enhance event extraction in low-resource contexts, the existing methods have two shortcomings that need further improvement. First, existing machine reading comprehension methods model the Q&A task in a single-round Q&A model, ignoring the valid information generated in the previous round of Q&A, while a multi-round Q&A process can effectively retain and utilize the a priori information learned in each round of Q&A pairs to gradually obtain the entities required in the next round, further achieving the effect of enhancing semantic information. Second, the existing model extracts independently for each element of the thesis element in the event, ignoring the dependency relation between the thesis elements, and introducing the information of historical answers can effectively tap into the correlation between different thesis elements and improve the performance of the model. Consider the following example sentence.

Example 2. *May 4 (Xinhua), British anti-terror police arrested Sharif in London on May 2.*

In the sentence of Example 2, if a single-round question and answer are used, the lack of semantic information at the multi-round level will lead to ambiguity in entity recognition when extracting temporal argument information for “May 4” and “May 2”. However, if “British anti-terror police” as the arresting party (Arrest-Jail, Agent) and “Sharif” as the arrested person (Arrest-Jail, Person) in the event were extracted during the previous rounds of questions and answers, they could be integrated into the subsequent rounds of questions, which could be disambiguated across sentences and effectively extract the time element “May 2” (Arrest-Jail, Time) in the event of arrest and imprisonment accurately. Therefore, building a multi-round Q&A framework helps the performance of the event extraction task.

To address the above shortcomings, this paper investigates how to introduce valid information into the Q&A framework to further enhance the performance of the event extraction model based on the existing machine reading comprehension model. Specifically, this paper extends and refines the model input and model framework, respectively, and proposes an event extraction method based on a multi-round Q&A model.

The main contributions of this paper include the following three points:

- (1) In terms of model input, this paper constructs a multi-round problem template for event argument elements extraction, which enables the model to effectively use prior knowledge to learn more adequate semantic information for additional performance gains.
- (2) In terms of the model framework, this paper introduces historical answer information into the multi-round question-and-answer framework, which is populated into the corresponding slots of the question rounds, enabling the model to effectively capture the correlations and hierarchical dependencies among the argument entities.
- (3) A multi-round Q&A model based on a machine reading comprehension model for event extraction is proposed in combination with the above two aspects, and its effectiveness is verified on a public dataset.

2. Related Work

The current event extraction methods proposed by scholars at home and abroad are mainly classified as the following: pattern-matching based [4], machine-learning based [5], and deep-learning based [6].

Both event extraction and relation extraction are initially performed based on template matching, where templates are obtained manually or automatically and extracted by various template matching algorithms to find the information that meets the template constraints. In 2010, Liao et al. [7] proposed the cross-event model to enhance the performance of multi-event type extraction systems by using cross-event document-level information for pattern matching in the face of the fact that most of the event extraction systems at that time were based on phrase- or sentence-level extraction. The RBRB model of Sha et al. 2016 [8] uses trigger words, sentence representations, and pattern features as inputs to the trigger word recognition task and introduces regularization methods to extract trigger words, allowing the model to use both pattern-based and representation-based information, and the classification performance of the model is significantly improved. However, it takes a lot of time for professionals to build the template, which is too costly and has relatively poor portability.

The event extraction method based on machine learning [9] is actually feature selection of textual information, constructing of binary or multivariate classification by machine learning algorithms, and event detection and argument extraction by classification. Machine-learning-based event extraction methods can usually eliminate non-event sentences from the text and use multiple knowledge fusion to represent candidate event instances. For example, Liao et al. [7] used information from other events in the same text as feature information in the event extraction process for training and obtained better recognition accuracy. However, machine learning methods mainly rely on feature engineering and analysis, which require the high professional ability of researchers and are more

time-consuming and laborious, and machine learning-based methods are prone to data sparsity problems, so there are still limitations.

With the advancement of technology and the rapid development of deep neural networks [10], deep learning techniques have been widely used for modeling complex structures and verified to be effective for many NLP tasks, and event extraction based on methods such as convolutional neural networks (CNN) [11], recurrent neural networks (RNN), and graphical neural networks (GNN) have become a hot research topic. In 2018, Sha et al. [12] proposed the dbRNN model, a bridge-dependent recurrent neural network, which is essentially based on a bidirectional RNN for event extraction, relying on relational graph information to extract event triggers and elemental roles. In 2019, Nguyen et al. [1] proposed a JMEE model based on graph convolutional networks, which relies on entity mentions aggregating convolutional vectors. The graph-based convolutional vectors are merged for the entities mentioned in the current word and sentence to improve the performance of event detection. In 2019, Nguyen et al. [2] also proposed Joint3EE, a joint task model for deep learning, to combine named entity recognition [13], trigger word recognition, and argument recognition, where three tasks share hidden representations, enabling knowledge sharing among tasks and capturing dependencies and interactions between tasks, thus improving the performance of event extraction. In 2019, Yang et al. [14] proposed the PLMEE model, which uses a pre-trained language model to extract event trigger words, a role prediction separation method to deal with role overlap in role extraction tasks and use the masked LM of the Bert [15] model to automatically generate labeled data with excellent results. Although the pre-trained model learns deeper interaction information and thus reduces extraction errors, the approach ignores the a priori semantic information of element descriptions. In 2020, Du et al. [16] first proposed to convert trigger word extraction and argument extraction into corresponding question and answer questions to complete event extraction with a question-and-answer task model, which no longer relies excessively on entity class information obtained from upstream tasks, thus reducing the problem of error propagation in event detection tasks. It is confirmed that Q&A and MRC are more advantageous compared to the traditional sequence annotation methods. However, this method ignores the a priori information in historical question-and-answer pairs and the hierarchical dependence of question-and-answer patterns. In this paper, we will build an event extraction model based on multi-round question-and-answer patterns based on this study to further improve the performance of the overall model.

3. Models

This section proposes an event extraction method based on a multi-round Q&A framework to address the shortcomings of existing methods that do not fully utilize a priori information and question-and-answer hierarchical relevance. The overall flow of the method is first outlined, and then the specific implementation is described, including the method of constructing question templates in a multi-round Q&A framework, the strategy of embedding historical answers into question-filling slots, and the method of marking event information in question construction. The overall framework of the model is shown in Figure 1.

3.1. Event Extraction Process Based on Multi-Round Q&A Framework

In this paper, we model the event detection task and the argument element extraction task as a multi-round Q&A framework via a pipeline model [17]. The complete multi-round flow of event extraction in this model is given in Figure 2. After inputting a sentence, the trigger word <Trigger> is first extracted from the sentence by the trigger word question-and-answer model. The extracted trigger words and their locations are input into the trigger word classification Q&A template, and the corresponding event type <Event Type> is extracted. Based on the obtained event type, the thesis element role <Event Type; role> is introduced into the construction of the thesis element question template, and the entity corresponding to the thesis element role is extracted. Then combine the answer

Deep NN

Graph based convolution
neural networks

PLMEE

advantage of this model

Multi-round QA model

entity information into the next round of question templates and further extract the other argument element roles of that event type. The above process is repeated to extract the remaining argument elements until the extraction of all event information is completed. The whole extraction process constitutes a machine reading comprehension task in a multi-round question-and-answer framework, which can flexibly model the event detection and argument element extraction tasks to achieve efficient event extraction.

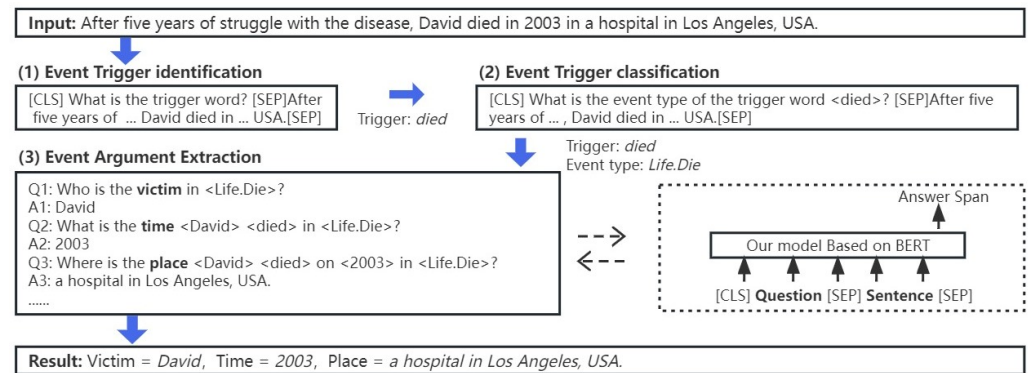


Figure 1. The overall framework of the model.

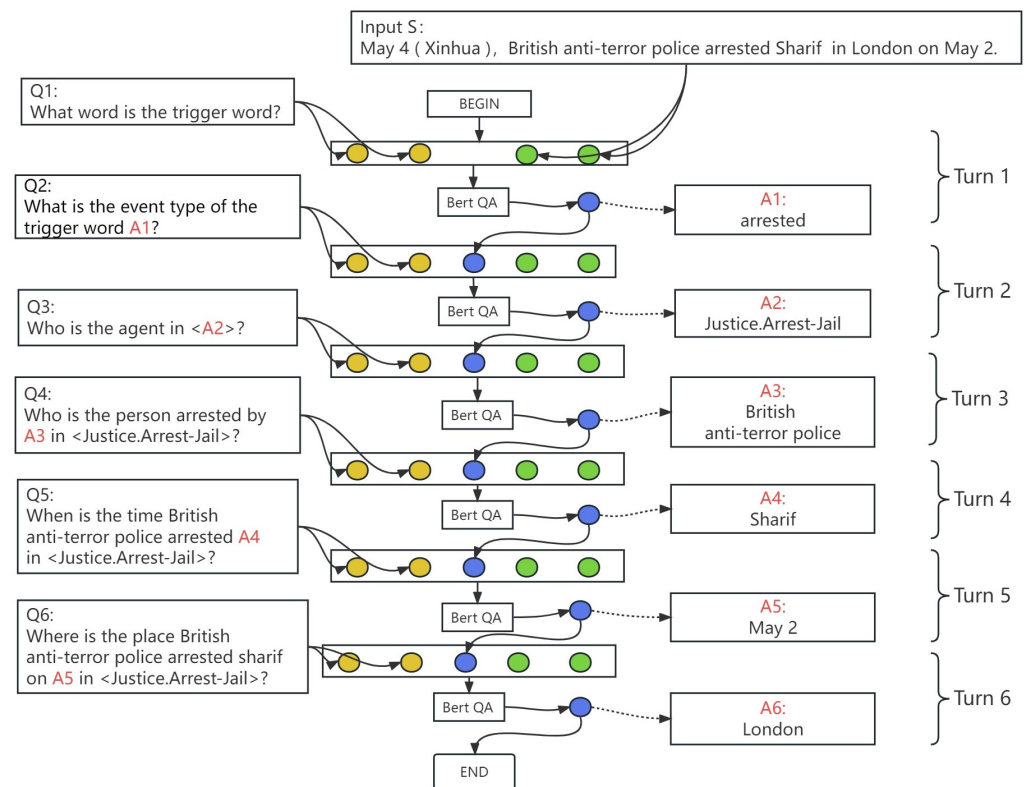


Figure 2. Complete multi-round flow for event extraction.

3.2. The Construction Method of Question Templates in a Multi-Round Q&A Framework

In the multi-round Q&A process of this paper, the construction of questions is a key step, and questions that fully incorporate a priori information help to enhance learning to more semantic information. Before defining the questions, this paper analyzed the ACE2005 dataset [18], which consists of 8 event types and 33 subtypes. Within each event type, a set of potential participant roles is included for entities that appear within its paradigm. In some cases, the question of whether a potential event is taggable depends on the presence of entities that populate certain roles. Among the event argument elements, there are

two main categories of argument elements: event participants and event attributes. An event participant is a markable entity that participates in some way in an event in a certain class of events. For example, <Person> in a marriage event <Marry>; <Agent>, <Victim>, <Instrument> in a death event <Die>; and so on. Event attributes are attributes associated with events other than participants, such as <Time>, <Place>, <Position>, and so on. Table 1 shows the event types and parameters in the ACE2005 dataset.

Table 1. Event types and parameters in ACE2005 dataset.

Event Type	Type	Event Role
Life	Be-Born	person, time, place
	Marry	person, time, place
	Divorce	person, time, place
	Injure	time, place, agent, victim, instrument
	Die	time, place, agent, victim, instrument
Movement	Transport	time, transporter, artifact, vehicle, price, origin, destination
Transaction	Transfer-Ownership	time, place, buyer, seller, beneficiary, artifact, price
	Transfer-Money	time, place, giver, recipient, beneficiary
Business	Start-Org	time, place, agent, org
	Merge-Org	time, place, org
	Declare-Bankruptcy	time, place, org
	End-Org	time, place, org
Conflict	Attack	time, place, attacker, target, instrument
	Demonstrate	time, place, entity
Contact	Meeting	time, place, entity, duration
	Phone-Write	time, entity
Personnel	Start-Position	person, time, place, entity, position
	End-Position	person, time, place, entity, position
	Nominate	person, time, place, agent, position
	Elect	person, time, place, agent, position
Justice	Arrest-Jail	person, time, place, agent, crime
	Release-Parole	person, time, place, entity, crime
	Trial-Hearing	time, place, defendant, prosecutor, adjudicator, crime
	Charge-Indict	time, place, defendant, prosecutor, adjudicator, crime
	Sue	time, place, defendant, adjudicator, crime, plaintiff
	Convict	time, place, defendant, adjudicator, crime
	Sentence	time, place, defendant, adjudicator, crime, sentence
	Fine	time, place, entity, adjudicator, crime, money
	Execute	person, time, place, agent, crime
	Extradite	person, time, agent, crime, destination, origin
	Acquit	time, place, defendant, adjudicator, crime
	Pardon	time, place, defendant, adjudicator, crime
	Appeal	time, place, defendant, adjudicator, crime, prosecutor

The “wh-” question word used in the question template is defined according to the different attributes of the argument, and the question is constructed using the [wh_word]. If the role type is <Person> and the participants of various activities, the question is constructed with the question word “Who”. If the role type is <Place>, the corresponding “wh-” word is “Where”, and if the role type is <Time>, the question is “When”. If the role type is a generic semantic role, the corresponding “wh-” question word is “What”. The specific correspondence is shown in Table 2.

Table 2. Correspondence between wh-words and event roles.

WH_WORD	Event Role
Who	Adjudicator, attacker, beneficiary, buyer, seller, victim, entity, transporter, Defendant, person, plaintiff, prosecutor, recipient, giver, agent, target
Where	Destination, place, origin
When	Time, duration
What	Instrument, artifact, vehicle, crime, money, price, sentence, org, position

After determining the question words in the question template, to be more linguistically logical, the name of the argument role of the corresponding event and the event type information is also introduced as part of the question. In this way, the question is accompanied by the semantic information of the event type, which makes the semantic information more complete and rich. Take the arrest and imprisonment event as an example. Table 3 shows the multi-round question and answer template constructed in this paper for this event.

Table 3. Multi-round Q&A Template.

Category	Template of the Question
Trigger	What is the trigger word?
Event Type	What is the event type of the trigger word A1?
Event Argument	Who is the agent in <A2>?
	Who is the person A1 by A3 in <A2>?
	When is the time A3 A1 A4 in <A2>?
	Where is the place A3 A1 A4 on A5 in <A2>?

First, obtain the trigger word Trigger, labeled A1 (Answer 1), in the first round of the trigger word extraction problem template. In the second round event classification problem template, A1 is introduced as part of the problem to obtain the Event type, labeled A2 (Answer 2). After determining the trigger word and event type, the extraction of the event argument element begins in the third round, and the question is constructed using the name of the argument role and the question word corresponding to the role in that event type, and the Event type (A2) is encoded to the end of the question template to extract the argument role Agent, labeled A3 (Answer 3). In the fourth round, the historical round answers A1 (Trigger), A2 (Event type), and A3 (Agent) are filled into the corresponding slots, and the thesis element Person is extracted, marked as A4 (Answer 4). Repeat the above process to extract the remaining argument element roles according to the corresponding question template. Finally, the extraction of the event information is completed.

3.3. History Answer Embedding Strategies

In this paper, we adopt a history answer embedding strategy, the basic idea of which is to use the answers of the previous round to construct the questions of the next round in a multi-round Q&A. The specific process is as follows: construct a multi-round question template, set the positions in the template as several blank slots for filling different contents in different rounds, obtain the historical answers of the previous round, input the marked historical answers and the corresponding marked positions in the sentences into the question template of the next round, and match the historical answers with the slots in the question template. The matching results are filled into the corresponding blank slots to generate the questions for the next round, which are fed into the answer extraction model to get the answers for that round, and the above process is repeated until the end of extraction.

Specifically, the general steps to implement the history answer embedding strategy in the model are: first, the history answer is encoded to obtain the History Answer Embedding, which is embedded into the model as an additional input and spliced with the current question template to form a new input sequence to obtain the final input representation. By embedding the history answer into the model, it enables the model to make better use of the history answer information, thus improving the accuracy of event extraction.

In the model, the answers extracted from the upstream rounds affect not only their own accuracy but also the construction of downstream questions, and the accuracy of each round of Q&A also has an impact on the performance of subsequent rounds. To address this issue, the multi-round Q&A framework needs to select an optimal text span to answer the questions. To select the text span, a strategy needs to be defined that determines the probability distribution of each span of text based on the current question and contextual information to improve the efficiency of the event extraction task. This algorithm is based on the output of the BMES tag, which assigns the probability of selecting spans to B (beginning), M (middle), and E (end) in the tag, respectively. Specifically, probability $\{w_1, w_2, \dots, w_n\}$ of selecting a span is w_1 assigned to the joint probability [19] of B, w_2, \dots, w_{n-1} is assigned to the joint probability of all M tags including the selected span, and probability w_n is assigned to the joint probability of E, i.e., as shown in Equation (1):

$$p(y(w_1, \dots, w_n) = \text{answer} | \text{question}, s) = p(w_1 = B) \times p(w_n = E) \prod_{i \in [2, n-1]} p(w_i = M) \quad (1)$$

In addition, an empirical replay strategy [20] is used for each training batch. That means half of the samples are set to be simulated while the other half are randomly selected from the previously generated dataset. It helps to increase the diversity of the dataset and improve the generalization of the model and reduce the overfitting of the model, further improving the training efficiency and the performance of the model.

3.4. Event Information Markers in Issue Construction

In the above multi-round Q&A framework, after obtaining the trigger words and event types from the trigger word identification and classification model, for each template type, the identified event type and the token position of the event type are entered into the multi-round question template by adding “in <Event type>” at the end of the question template extracted from the argument elements. The question Q and sentence S are then entered into the argument element extraction model. The following Equation (2) is the base template containing the event type information:

$$< WH_word > \text{ is the } < event \text{ role } > \text{ in } < event \text{ type } >? \quad (2)$$

where <event type> is instantiated from the real event type token obtained during the trigger detection task and is intended to introduce the event type as a priori information to construct the problem to be used as an indicator of the information in the input sentence against the event argument elements.

In the example shown in Figure 3, the event type < Justice.Arrest-Jail> extracted in the previous round is added to the end of question Q before feeding it into the model. By explicitly annotating the event type in the question template, on the one hand, the associated parameter roles can be quickly identified, and, on the other hand, it allows the model to effectively use the prior event information in the learning process, thus better assisting in the extraction of event information.

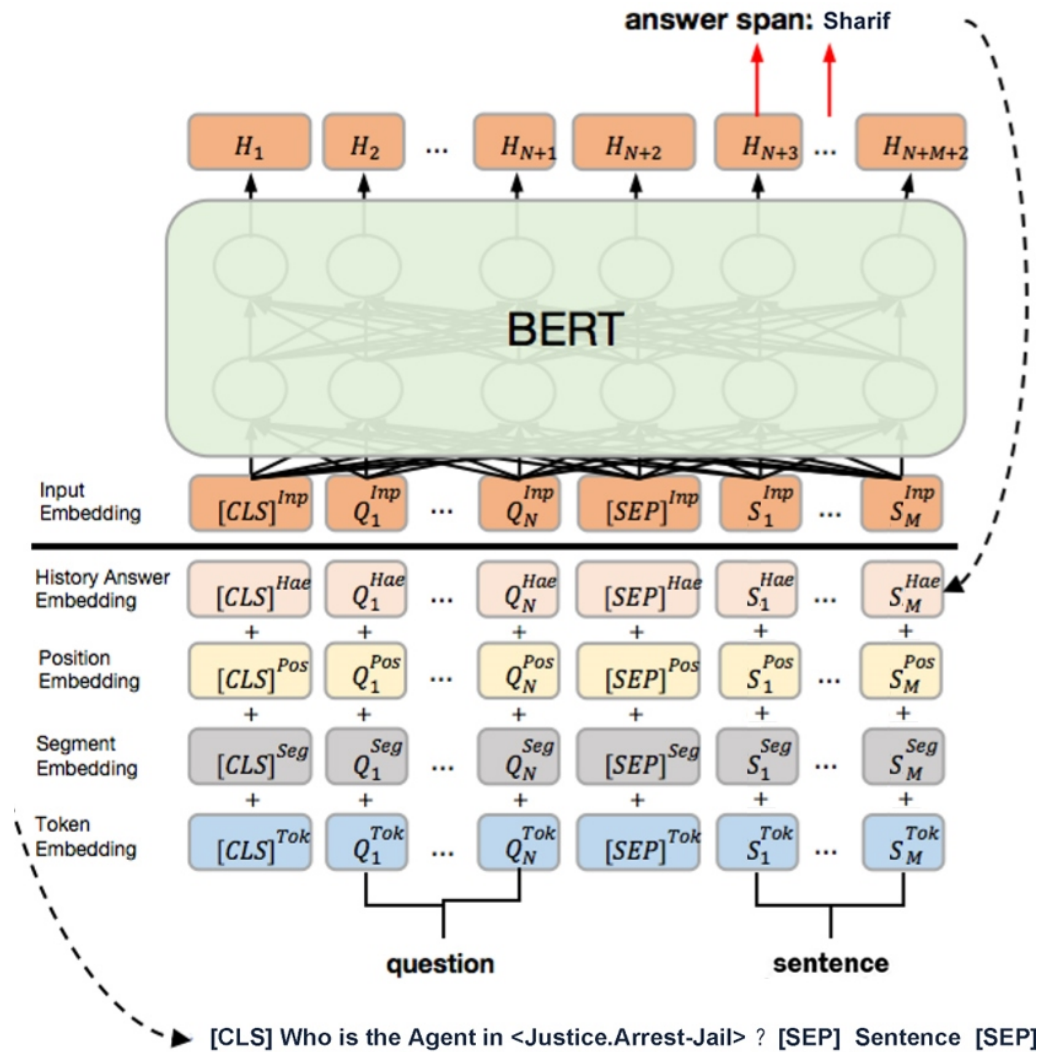


Figure 3. Event extraction process based on multi-round Q&A framework.

3.5. Multi-Round Answer Extraction Model

In this paper, two answer extraction models, a trigger detection model, and an argument element extraction model are proposed separately using BERT as the base model to obtain contextual representations from their input sequences, and the argument elements are updated during the training process. After instantiation using the question template, the sequence is shown in Equation (3):

$$[CLS] \langle Question \rangle [SEP] \langle Sentence \rangle [SEP] \quad (3)$$

The contextual representation of each token used for trigger detection and argument extraction is obtained using $BERT_{Tr}$ and $BERT_{Arg}$, respectively. For the input sequence (a_1, a_2, \dots, a_N) prepared for triggering detection, we have Equations (4) and (5):

$$A = [a_1, a_2, \dots, a_N] \quad (4)$$

$$a_1, a_2, \dots, a_N = \text{BERT}_{Tr}(a_1, a_2, \dots, a_N) \quad (5)$$

As for the input sequence (b_1, b_2, \dots, b_M) prepared for the argument element span extraction, we have Equations (6) and (7):

$$B = [b_1, b_2, \dots, b_M] \quad (6)$$

$$b_1, b_2, \dots, b_M = \text{BERT}_{Arg}(b_1, b_2, \dots, b_M) \quad (7)$$

However the output layer of each QA model is different: the trigger detection model predicts the event type of each token in the sentence, while the argument extraction model predicts the start and end offsets of the argument span.

For trigger prediction, this paper introduces a new parameter matrix $W_{QAttr} \in R^{D \times H}$, where D is the hidden size of the transformer, H is the number of event types plus one (for non-trigger tokens), and the softmax normalization applies to H types, as shown in Equation (8):

$$P_{QAttr} = \text{softmax}(EW_{QAttr}) \in R^D \times N \quad (8)$$

For the argument element prediction, two new parameter matrices $W_{QAarg1} \in R^{D \times 1}$ and $W_{QAarg2} \in R^{D \times 1}$ are introduced in this paper. Perform softmax normalization on the input tokens c_1, c_2, \dots, c_M to obtain the probability of each token being selected as the start/end of the range of argument span, as shown in Equations (9) and (10):

$$P_{QAarg1}(i) = \text{softmax}(c_i W_{QAarg1}) \quad (9)$$

$$P_{QAarg2}(i) = \text{softmax}(c_i W_{QAarg2}) \quad (10)$$

To train the model, the log-likelihood loss of both models is minimized in this paper. In particular, the loss of the argument extraction model is the sum of two components: the start token loss and the end token loss. For the training example with no parameter span (no answer case), we minimize the start and end probabilities of the first token of the sequence ([CLS]), as shown in Equation (11):

$$\zeta_{QAarg} = \zeta_{arg_start} + \zeta_{arg_end} \quad (11)$$

To obtain the type of each token a_1, a_2, \dots, a_N at test time for trigger detection, simply apply argmax to P_{QAttr} .

4. Experiment

4.1. Dataset

This paper conducts model evaluation experiments on the ACE 2005 corpus, which the corpus contains English, Arabic, and Chinese training datasets used for the evaluation of the 2005 Automatic Content Extraction (ACE) review conference. The corpus consists of various types of data annotated by the Language Data Consortium (LDC) for entities, relations, and events, including documentary news crawled from multiple domains such as newsletters, weblogs, and broadcast conversations. It contains 599 English documents, defining 8 event types, 33 subtypes, and 35 argument role types. This paper is divided into training set, validation set, and test set according to a ratio of 7:2:1.

4.2. Evaluation Metrics

In this paper, following the evaluation metrics used by previous researchers, The trigger is correctly identified if the span offset of the trigger exactly matches the reference trigger; the trigger is correctly classified if the span offset and event sub-type of the trigger exactly match the reference trigger; the parameter is correctly identified if its span offset and corresponding event sub-type exactly match the referenced parameter; and the parameter is correctly classified if its span offset, corresponding event sub-type, and parameter role exactly match those of the referenced parameter.

For each of the above metrics, performance is evaluated using P (precision), R (recall), and F1 scores.

The specific formula for these three evaluation indicators is as follows:

$$P = \frac{TP}{TP + FP} \quad (12)$$

$$R = \frac{TP}{TP + FN} \quad (13)$$

$$F1 = \frac{2 \times P \times R}{P + R} \quad (14)$$

In the above equation: TP (true positive) denotes the number of true positive cases predicted to be positive; FP (false positive) denotes the number of non-true positive cases predicted to be positive; FN (false negative) denotes the number of true positive cases predicted to be negative.

In the task of this paper, the precision P is used to measure the accuracy of the model, the recall R is used to measure the comprehensiveness of the model, and the F1 value is the summed average of the precision and recall, which is used as an overall evaluation index of the model performance.

4.3. Experimental Settings

The pre-trained model used in this experiment is the Bert benchmark version (Bert-base-uncased), using the Pytorch framework to implement the proposed event extraction model, and the training process uses the Adam [21] optimizer to optimize the model, with Epoch set to 10 and batch_size value to 16. The learning_rate of the pre-training model is set to 2×10^{-5} , and the learning rate is dynamically adjusted using linear warm_up, taking the first ten percent of the steps as the peak division, and the learning takes the lead to rise and then fall, and the trigger word recognition extraction model and event element extraction model are trained for evaluation respectively.

4.4. Cross-Domain Knowledge Migration Experiments

In this paper, we utilize a multi-round model framework for event extraction based on machine reading comprehension models, one of the advantages of which is that the models can be pre-trained with existing large-scale machine reading comprehension datasets. In this paper, the Squad2.0 dataset [22] is used for pre-training, the trained model is saved, the model is fine-tuned on the dataset ACE2005 for the event extraction task, and the training is completed and tested on the test set, a process equivalent to cross-domain knowledge-transfer learning. Therefore, this section designs a set of experiments to verify the effectiveness of cross-domain knowledge migration in the event extraction task.

As shown in Tables 4 and 5, the experimental results of the model without cross-domain knowledge pre-training (Ours_NoPre) and the model after cross-domain knowledge pre-training (Ours) are presented for the trigger word recognition extraction task and the event element recognition extraction task, respectively.

Table 4. Trigger word recognition and classification results based on cross-domain knowledge pre-training.

Methods	Trigger Identification (%)			Trigger Identification + Classification (%)		
	P	R	F1	P	R	F1
Ours_NoPre	72.10	75.02	73.53	70.43	71.38	70.90
Ours	75.32	78.26	76.76	74.19	74.22	74.20

From the above experimental results, it can be seen that with the same evaluation metrics used, the F1 scores improved after introducing the pre-trained model (Ours) from the cross-domain dataset, both for the trigger word recognition and classification task and the event element recognition and classification task. Specifically, for the trigger word recognition and classification tasks, the F1 values improved by 3.3%, while in the event element recognition and classification tasks, the F1 values improved by more than 3.6%. The experimental results strongly demonstrate the effectiveness of cross-domain knowledge transfer and also validate the advantages and value of using a multi-round pattern framework for event extraction based on machine reading comprehension models in this paper.

Table 5. Recognition and classification results of event elements based on cross-domain knowledge pre-training.

Methods	Argument Identification (%)			Argument Identification + Classification (%)		
	P	R	F1	P	R	F1
Ours_NoPre	65.68	60.17	62.80	64.42	54.27	58.91
Ours	69.60	61.52	65.31	67.67	58.43	62.71

4.5. Comparison Experiments

In order to verify the overall performance of the model, several models that have achieved better experimental results in recent years are used in this section as a benchmark for comparison with the model in this paper. The specific model is described as follows:

- **JointBeam**: A classical event extraction algorithm based on traditional machine learning using manually designed features based on structured predictions to extract events. In this paper, we select the model as a control group for feature-engineering-based event extraction methods;
- **dbRNN**: A bi-directional long- and short-term memory network based event extraction method. utilizing the method of adding relational dependency bridges to the BiLSTM for event extraction. This method is more representative before the advent of pre-trained models and is often used as a baseline in the same type of studies. Therefore, this model is selected as a control group for non-machine reading-comprehension-based event extraction methods in this paper;
- **BERT_QA**: An event extraction based on question and answer model, which is an earlier proposal to construct the event extraction task as a question and answer (QA)/machine reading comprehension (MRC) task. In this paper, the model is extended on the basis of this model, so it is selected as a control group;
- **MQAEE**: A bert-based multi-round Q&A type event extraction. The idea of multi-round Q&A in this method has similarity with this paper, but its multi-round Q&A is framed as independent rounds of Q&A, and the dependency between argument elements is not fully considered, so the model has been selected as a control group against the effectiveness of the historical answer information embedding strategy in this paper.

The final experimental results obtained are shown in Tables 6 and 7, which list the experimental results of comparing the algorithm in this paper with other benchmark methods in the trigger word recognition extraction task and the event element recognition extraction task.

From the above experimental results, the following three conclusions can be drawn:

(1) Comparing the neural-network-based event extraction approach (dbRNN) and the feature-engineering-based approach (JointBeam), it can be found that the neural-network-based model has a clear advantage because the neural network can better learn the semantic feature representation of the sentence, so it can achieve better performance.

(2) Comparing non-machine reading-comprehension-based event extraction methods (dbRNN) and machine reading-comprehension-based methods (BERT_QA, MQAEE), it can be found that models based on machine reading comprehension methods outperform those based on classification or sequence annotation because the former can encode a priori information in the problem, allowing the model to focus more on the extraction of specific information in the passage, thus effectively improving the performance of event extraction.

(3) Comparing this paper's model (Ours) with similar methods (BERT_QA, MQAEE), this paper's method achieves the best performance on two subtasks: trigger word recognition and classification, and event element recognition and classification. For trigger word recognition and classification, the F1 value of the model in this paper is 74.2%, which is better than all benchmark methods. The model in this paper also has an excellent performance in event element recognition and classification, with an F1 value of 62.71%. Compared with the second-ranked dbRNN method in the benchmark method, the F1 value of the method in this paper is improved by more than 4%. It is illustrated that constructing a multi-round Q&A framework and embedding historical answer information encoding enables the model to effectively capture the correlations and hierarchical dependencies among argument entities, enhancing the final results and validating the effectiveness of the method in this paper for the event extraction task.

Table 6. Comparison experimental results of trigger word recognition and classification.

Methods	Trigger Identification (%)			Trigger Identification + Classification (%)		
	P	R	F1	P	R	F1
JointBeam	76.90	65.00	70.40	73.70	62.30	67.50
dbRNN	-	-	-	74.10	69.80	71.90
BERT_QA_Trigger	74.29	77.42	75.82	71.12	73.70	72.39
MQAEE	-	-	-	-	-	73.80
Ours	75.32	78.26	76.76	74.19	74.22	74.20

Table 7. Comparison experimental results of event element recognition and classification.

Methods	Argument Identification (%)			Argument Identification + Classification (%)		
	P	R	F1	P	R	F1
JointBeam	69.80	47.90	56.80	64.70	44.40	52.70
dbRNN	71.30	64.50	67.70	66.20	52.80	58.70
BERT_QA_Arg	58.90	52.08	55.29	56.77	50.24	53.31
MQAEE	-	-	-	-	-	55.00
Ours	70.50	67.41	68.92	67.67	58.43	62.71

4.6. Ablation Experiments

To verify the effectiveness of embedding event information markers in the multi-round Q&A templates in this paper, this section further designs ablation experiments to compare the performance changes of the model before and after removing the event information markers. The experimental results are listed in Table 8.

Table 8. Results of ablation experiments.

Methods	Trigger Word Identification and Classification (%)			Event Argument Identification and Classification (%)		
	P	R	F1	P	R	F1
w/o Event information mark	71.80	72.81	72.30	65.32	56.34	60.50
Ours	74.19	74.22	74.20	67.67	58.43	62.71

According to the experimental results in Table 8, it can be observed that the performance of trigger word recognition and classification as well as event element recognition and classification has a good performance improvement after embedding event information tags. Specifically, there is an improvement of more than 1.9% in the F1 value for trigger word recognition and classification and a significant improvement of more than 2% in the F1 value for event element recognition and classification. The experimental results show that introducing event information tagging into the session encoding module helps to improve the performance of the event extraction model.

4.7. Case Studies

The following two cases are given in this section. To better illustrate the effectiveness of the model in the event extraction task, Case 1 uses an example sentence from an event in the dataset:

Example: “Mark was born in New York three years ago.”

The specific problem process is shown in Figure 4:

Elements	Question generation	Extraction results
Trigger	[CLS] What is the trigger word? [SEP] Mark was born in New York three years ago.[SEP]	“was born”
Event Type	What is the event type of the trigger word <was born>?	Life.Be-Born
Args.Person	Who is the person in <Life.Be-Born>?	“Mark”
Args.Time-within	What is the time <Mark><was born>in <Life.Be-Born>?	“three years ago”
Args.Place	Where is the place <Mark><was born> on <three years ago> in <Life.Be-Born>?	“New York”

Figure 4. Case 1.

In Case 2, we further verify the validity of the methodology of this paper from the perspective of qualitative analysis.

The sentence shown in Figure 5 contains an event (Justice. Arrest-Jail), in which the event is triggered by the trigger word “arrested”, and P1 represents the prediction result of the model based on a multi-round Q&A framework incorporating historical answer information. P2 denotes the model prediction result in a multi-round independent question-and-answer framework. For the event detection task, both models correctly identified the trigger word “arrested” and determined the event type to be “Arrest-Jail”. Only the model using the multi-round Q&A framework correctly identified the temporal element “May 2” of the Arrest-Jail event in the element identification and classification task, while the model not using the multi-round historical question-and-answer framework incorrectly identified the element.

Input: May 4 (Xinhua), British anti-terror police arrested Sharif in London on May 2.	
P1	P2
Q1: What is the trigger word?	Q1: Which word is the trigger word?
A1: arrested	A1: arrested
Q2: What is the event type of the trigger word arrested?	Q2: Which is the event type of the trigger word arrested?
A2: Justice.Arrest-Jail	A2: Justice.Arrest-Jail
Q3: Who is the agent in <Justice.Arrest-Jail>?	Q3: Who is the agent in <Justice.Arrest-Jail>?
A3: British anti-terror police	A3: British anti-terror police
Q4: Who is the person arrested by British anti-terror police in <Justice.Arrest-Jail>?	Q4: Who is the person in <Justice.Arrest-Jail>?
A4: Sharif	A4: Sharif
Q5: When is the time British anti-terror police arrested sharif in <Justice.Arrest-Jail>?	Q5: When is the time in <Justice.Arrest-Jail>?
A5: May 2	A5: May 4
Q6: Where is the place British anti-terror police arrested sharif on May 2 in <Justice.Arrest-Jail>?	Q6: Where is the place in <Justice.Arrest-Jail>?
A6: London	A6: London

Figure 5. Case 2.

5. Conclusions

In this paper, we take event extraction as the research object. In response to the shortcomings of existing methods that do not utilize a priori information in historical question-and-answer pairs and do not explore the hierarchical dependency of multi-round question-and-answer, an event extraction method based on a multi-round Q&A model is proposed. The approach refines the existing method in two ways: First, constructing a multi-round question-and-answer framework enables the model to effectively augment semantic information with prior knowledge. Second, by encoding historical answer information into the question template, the correlation between argument elements is used to achieve aided reasoning for the event extraction task.

Author Contributions: Conceptualization, L.H., X.Z., L.Z. and Q.Z.; writing—original draft preparation, L.H. and X.Z.; writing—review and editing, L.H.; data curation, X.Z.; validation, L.Z.; supervision, L.Z., L.H. and Q.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the RD Program of the Beijing Municipal Education Commission (KM202210009002), the National Natural Science Foundation of China (61972003, 62007027), and the Beijing Urban Governance Research Center Fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank the anonymous reviewers for their helpful comments. We would like to thank the referees for their comments, which helped improve this paper considerably.

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

References

1. Nguyen, T.; Grishman, R. Graph convolutional networks with argument-aware pooling for event detection. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 7370–7377.
2. Nguyen, T.M.; Nguyen, T.H. One for all: Neural joint modeling of entities and events. In Proceedings of the AAAI conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 6851–6858.
3. Lafferty, J.; McCallum, A.; Pereira, F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of the International Conference on Machine Learning, San Francisco, CA, USA, 28 June–1 July 2001; pp. 282–289.
4. Patil, N.V. An Emphatic Attempt with Cognizance of the Marathi Language for Named Entity Recognition. *Procedia Comput. Sci.* **2023**, *218*, 2133–2142. [\[CrossRef\]](#)
5. Gultiaev, A.A.; Domashova, J.V. Developing a named entity recognition model for text documents in Russian to detect personal data using machine learning methods. *Procedia Comput. Sci.* **2022**, *213*, 127–135. [\[CrossRef\]](#)
6. Oliveira, B.S.N.; de Oliveira, A.F.; de Lira, V.M.; da Silva, T.L.C.; de Macedo, J.A.F. HELD: Hierarchical entity-label disambiguation in named entity recognition task using deep learning. *Intell. Data Anal.* **2022**, *26*, 637–657. [\[CrossRef\]](#)
7. Liao, S.; Grishman, R. 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, 11–16 July 2010; pp. 789–797.
8. Sha, L.; Liu, J.; Lin, C.Y. RBPB: Regularization-based pattern balancing method for event extraction. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 1224–1234.
9. Zeiler, M.D.; Fergus, R. Stochastic pooling for regularization of deep convolutional neural networks. *arXiv* **2013**, arXiv:1301.3557.
10. Yang, T.; He, Y.; Yang, N. Named Entity Recognition of Medical Text Based on the Deep Neural Network. *J. Healthc. Eng.* **2022**, *2022*, 3990563. [\[CrossRef\]](#)
11. Li, Y.; Chen, F. Convolutional Neural Networks for Classifying Cervical Cancer Types Using Histological Images. *J. Digit. Imaging* **2022**, *36*, 441–449. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Sha, L.; Qian, F.; Chang, B. Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 5916–5923.
13. Luo, Y.; Xiao, F.; Zhao, H. Hierarchical contextualized representation for named entity recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 8441–8448.
14. Yang, S.; Feng, D.; Qiao, L. Exploring pre-trained language models for event extraction and generation. In Proceedings of 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 29–31 July 2019; pp. 5284–5294.
15. Kenton, J.D.M.W.C.; Toutanova, L.K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
16. Du, X.; Cardie, C. Event extraction by answering (almost) natural questions. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, Online, 16–20 November 2020; pp. 671–683.
17. Chen, Y.; Xu, L.; Liu, K. 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, Beijing, China, 26–31 July 2015; pp. 167–176.
18. Wang, Y. Dependency multi-weight-view graphs for event detection with label co-occurrence. *Inf. Sci.* **2022**, *606*, 423–439. [\[CrossRef\]](#)
19. Zhu, E.Y.; Johri, S.; Bacon, D. Generative quantum learning of joint probability distribution functions. *Phys. Rev. Res.* **2022**, *4*, 043092. [\[CrossRef\]](#)
20. Mnih, V.; Kavukcuoglu, K.; Silver, D. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015; pp. 1–13.
22. Rajpurkar, P.; Jia, R.; Liang, P. Know what you don’t know: Unanswerable questions for SQuAD. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; pp. 784–789.

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