Neurocomputing 481 (2022) 121–132

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A multi-level neural network for implicit causality detection in web texts

Shining Liang [a,b], Wanli Zuo [a,b], Zhenkun Shi [b,c,*], Sen Wang [d], Junhu Wang [e], Xianglin Zuo [a,b,*]

[a] Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Changchun, China
[b] College of Computer Science and Technology, Jilin University, Jilin, China
[c] Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, China
[d] School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, Australia
[e] School of Information and Communication Technology, Griffith University, Queensland, Australia

ARTICLE INFO

Article history:
Received 4 July 2020
Revised 3 December 2021
Accepted 19 January 2022
Available online 24 January 2022

Keywords:
Causality Detection
Multi-level Neural Network
Relation Network
Transformer

ABSTRACT

Mining causality from text is a complex and crucial natural language understanding task corresponding to human cognition. Existing studies on this subject can be divided into two categories: feature engineering-based and neural model-based methods. In this paper, we find that the former has incomplete coverage and intrinsic errors but provides prior knowledge, whereas the latter leverages context information but has insufficient causal inference. To address the limitations, we propose a novel causality detection model named MCDN, which explicitly models the causal reasoning process, and exploits the advantages of both methods. Specifically, we adopt multi-head self-attention to acquire semantic features at the word level and develop the SCRN to infer causality at the segment level. To the best of our knowledge, this is the first time the Relation Network is applied with regard to the causality tasks. The experimental results demonstrate that: i) the proposed method outperforms the strong baselines on causality detection; ii) further analysis manifests the effectiveness and robustness of MCDN.

1. Introduction

Automatic text causality mining is an important yet challenging task because causality is believed to be critical in human cognition when making decisions [1]. Thus, automatic text causality has been extensively investigated in a variety of areas, such as medical [2], question answering [3] and event prediction [4], etc. A tool automatically extracts meaningful causal relations could help us construct causality graphs [5] to unveil previously undiscovered relationships between events and accelerate the discovery of the events' intrinsic logic [6].

Many research efforts have been made to mine causality from text corpora with complex sentence structures in books or newspapers [7–9]. However, the scale of textual data in the world, for example, on the web, is much larger than that in books and newspapers. Despite the success of prior studies on extracting explicit causality, the majority cannot be transferred directly to causality mining on the web text, which has a substantial number of implicit causality examples. First, most public available causality mining datasets are derived from books and newspapers. Their language expressions are typically formal but lack diversity than the web text. Second, the existing works mainly focus on explicit causal relations expressed by intra-sentence or inter-sentence connectives, omitting ambiguous and implicit instances. As is generally known, implicit causality always has a simple sentence structure without any connectives as below. In **Example 1**, "*got wet*" is the cause of "*fever*" and there are no connectives available for inference. Contrastively in **Example 2**, there are explicit connectives (i.e. "since" and "result") benefiting the causality detection.

- **Example 1:** I *got wet* during the day and came home with a *fever* at night.
- **Example 2:** Since computers merely execute the instructions they are given, *bugs* are nearly always the result of programmer *error* or an *oversight* made in the program's design.

As a result, it would make the recognition of causality incomplete if we ignore those implicit ones. There is an enormous demand to investigate an approach for mining both explicit and implicit causality from the web text.

In this paper, we formulate causality mining as two subsequential tasks: causality detection [4,10,11] and cause-effect pair extraction [12,13], which are also investigated in SemEval-2020 Task5 [14] and FinCausal 2020 [15]. When dealing with large-scale web text, detecting causalities is a prerequisite system before extracting cause-effect pairs. It could contribute to building a

* Corresponding authors.
E-mail address: shizk14@mails.jlu.edu.cn (Z. Shi).

https://doi.org/10.1016/j.neucom.2022.01.076
0925-2312/

high-quality corpus with diverse linguistic characteristics for causal pairs extraction, resulting in reduced annotation cost and model complexity for downstream tasks. In recent years, Hidey and McKeown [10] utilize the "AltLex" (i.e. Alternative lexicalization in the Penn Discourse Tree Bank) to build a large open-domain causality detection dataset based on parallel Wikipedia articles, therefore extending the scale of AltLex as shown in Table 1. Most existing works on the causality detection task falls into two categories: i) feature engineering-based methods, widely apply linguistic features (part-of-speech (POS) tagging, dependency parsing) [4,10] and statistic features (template, co-occurrence) [11]. There are thousands of AltLexes, some of which may appear as "consequently" in Table 1 in both causal and non-causal examples. However, the complicated features hardly capture the subtle discrepancies between various causality expressions, and the inherent errors of natural language processing (NLP) tools will be accumulated and propagated; ii) neural model-based methods, which have achieved notable results with the end-to-end paradigm, are prevalent in terms of design and usage. We conduct an empirical study to assess the application of neural text classifiers on causality detection. The performance of most neural model-based methods falls behind feature engineering-based methods (in Table 3). The explanation is encapsulated as they mainly focus on the interactions of the words and treat the sentence as a whole, but do not make explicit inferences about causality within sentences. Recently, pre-trained language models (PLMs) [16,17] develop dramatically and even surpass human performance on a variety of NLP tasks. Nonetheless, when dealing with large-scale web text data, the memory and time consumption is quite considerable.

Faced with the aforementioned problems, we propose the Multi-level Causality Detection Network (MCDN) for causality detection in web texts based on the following observations: i) neural network-based methods can reduce the labor cost and inherent errors of feature engineering-based methods while combining the prior knowledge of the latter benefits the former [18]; ii) causality reasoning is a high-level human ability [1] that necessitates multi-level analysis of the model. MCDN modifies a Transformer Encoder module to obtain semantic representation at the word level and combines a novel Self Causal Relation Network (SCRN) module at the segment level to infer causality via the segments on both sides of the connectives. Moreover, we claim that integrating multi-level knowledge may facilitate the token level feature overfitting proposed in [14].

Specifically, as shown in Fig. 1, MCDN splits the sentence into three segments based on the "segment before AltLex" (*BL*), "AltLex" (*L*), and "segment after AltLex" (*AL*). Intuitively, the cause and effect usually appear on both sides of the AltLex. This straightforward prior feature minimizes the impact of feature engineering complexity and errors. Motivated by explicitly modeling the causal reasoning process, the SCRN module encodes the segments and aggregates them into pair-wise groups, which are then concatenated with a sentence representation. Specifically, the interactions between the cause/effect segments and the connective, i.e., *BL-L* and *L-AL*, emphasize the role of the ambiguity of AltLex. It indicates if each segment conveys causality according to the current AltLex. For example, "made" is a causal connective in Fig. 1. However, the sentence "The baker made a cake" is non-causal. This illustrates that modeling the interactions between the AltLex and other segments is essential. Furthermore, by the interaction between causal segments, i.e., *B-A* and *A-B*, SCRN directly infers the potential causality when the two segments are coupled in a specific context. It stresses the role of segments co-occurrence, which also can be seen as we leverage the learnable parameters of the neural network instead of the statistical features. The above information constitutes the segment-level representation.

Next, we utilize Transformer architecture [19] at the word level. To maintain the framework fast and light in the large-scale web text scenario, the heads and blocks of the Transformer Encoder module are clipped and hence do not employ the pre-trained weights. Moreover, we extend the segment embedding to fit the multi-segment structure of the input. With this end-to-end module, MCDN combines local context and long-distance dependency to obtain a word-level representation. Finally, we perform detection with word-level and segment-level representations.

In general, the contributions of this paper can be summarized as below:

- First, we propose the MCDN framework for the causality detection task, which combines the advantages of feature engineering-based and neural model-based methods, which analyzes the causality within the sentence at multiple levels.
- Second, the relational reasoning module SCRN explicitly models the causal relations within sentences. By using this module, our method achieves a satisfying balance between performance and consumption compared with conventional neural classifiers and PLMs.
- Last, we conduct extensive experiments on the implicit causality detection dataset and take a counterfactual recognition dataset as a supplement. MCDN improves the SOTA performance of implicit causality detection and obtains competitive results on counterfactual recognition.

2. Related work

2.1. Causality mining

Datasets To date, the research interest in the causality of the community has increased gradually. Early attempts of SemEval benchmarks [20,21] formulates the causality detection as a relation classification task, i.e, given an entity pair with its context, the system needs to classify which relation the pair belongs to. As the development and evolution of routine events contain abundant causalities, there exists a series of works about event causality. In Causal-TimeBank [8], the authors introduce "CLINK" and "C-SIGNAL" tags to mark events causal relation and causal signals based on specific templates, respectively. Do et al. [7] collect 25 newswire articles from CNN in 2010 and release an event causality dataset that provides relatively dense causal annotations. Similarly aiming at extracting and classifying events relevant for stories, Caselli and Vossen [9] annotate a document-level corpus, Event StoryLine. Recently, Hidey and McKeown [10] expand the definition of "AltLex" to collect a large open-domain implicit causality detection dataset based on parallel Wikipedia articles, which is more sophisticated and noisy than the datasets above. Furthermore, without annotated symbols of the candidate pairs, the input of this task is raw text. It requires the models to perform fine-grained causal inference to figure out the discrepancy between causal and non-causal relation in the ambiguous or implicit context.

Methods Prior research can be roughly summarized into two primary categories: feature engineering-based and neural model-based. For feature engineering-based methods, a typical work [22] leverages dependency structure to extract cause-effect pairs. Zhao et al. [4] divide causal connectives into different classes as a new category feature based on the similarity of the syntactic dependency structure within causality sentences. Further studies incorporate world knowledge as a supplement to lexico-syntax analysis. Generalizing nouns to their hypernyms in WordNet and verbs to the parent classes in VerbNet [23–25] eliminates the negative effect of lexical variations and discovers frequent patterns of cause-effect pairs. Hidey and McKeown [10] incorporate world

Table 1
Hidey and McKeown [10] use the Simple Wikipedia as the parallel corpus to identify new connectives which are the paraphrases of the original AltLexes in the similar sentences of the English Wikipedia. The connective "instead" is a paraphrase of "consequently" and will be added to the new AltLexes.

AltLexes	English Wikipedia	Simple Wikipedia
Same	*A moving observer thus sees the light coming from a slightly different direction and consequently sees the source at a position shifted from its original position.*	*A moving observer thus sees the light coming from a slightly different direction and consequently sees the source at a position shifted from its original position.*
Paraphrase	*His studies were interrupted by World War I, and consequently taught at schools in Voronezh and Saratov.*	*However, he had to stop studying because of the World War I, and instead taught at schools in Voronezh and Saratov.*

Table 3
Results on AltLex.

Methods	Metrics			
	Accuracy	Precision	Recall	F1-score
Train set: *Training*				
MCC	63.50	60.32	**82.96**	69.85
KLD	58.03	**91.17**	19.55	32.20
LS ∪ *KLD*	73.95	80.63	64.35	71.57
LS ∪ *KLD* ∪ *CONN*	71.86	70.28	77.60	73.76
TextCNN	64.22 ± 0.68	71.13 ± 1.03	51.94 ± 3.13	59.73 ± 1.87
TextRNN	62.98 ± 0.34	70.33 ± 1.24	49.21 ± 2.15	57.71 ± 1.08
SASE	63.01 ± 0.18	64.92 ± 1.29	62.70 ± 3.78	63.36 ± 1.77
DPCNN	61.17 ± 0.60	61.41 ± 1.29	68.41 ± 4.03	64.24 ± 1.37
DRNN	64.09 ± 0.18	68.60 ± 1.40	56.89 ± 3.12	61.86 ± 1.25
MCDN	81.04 ± 0.81	82.28 ± 1.61	80.70 ± 0.87	81.47 ± 0.61
DistilBERT	67.59 ± 0.65	71.27 ± 0.63	62.22 ± 1.10	66.44 ± 0.81
BERT	65.79 ± 0.33	66.83 ± 1.07	67.09 ± 1.80	66.90 ± 0.39
Train set: *Bootstrapped*				
KLD ∪ *LS* ∪ LS_{inter}	79.58	77.29	84.85	80.90
TextCNN	74.83 ± 0.56	73.52 ± 1.45	80.51 ± 1.74	76.73 ± 0.29
TextRNN	75.22 ± 0.48	73.35 ± 0.76	81.74 ± 1.18	77.27 ± 0.44
SASE	70.02 ± 0.60	68.01 ± 0.88	79.30 ± 1.13	73.17 ± 0.36
DPCNN	76.66 ± 0.29	79.33 ± 1.21	74.29 ± 1.63	76.62 ± 0.41
DRNN	75.38 ± 0.18	73.33 ± 0.34	83.16 ± 0.71	77.48 ± 0.21
MCDN	80.75 ± 0.18	77.67 ± 0.36	88.63 ± 0.77	82.50 ± 0.19
DistilBERT	77.47 ± 0.74	75.52 ± 1.65	83.39 ± 1.43	79.24 ± 0.26
BERT	78.18 ± 0.55	75.89 ± 0.69	84.55 ± 0.42	79.98 ± 0.44

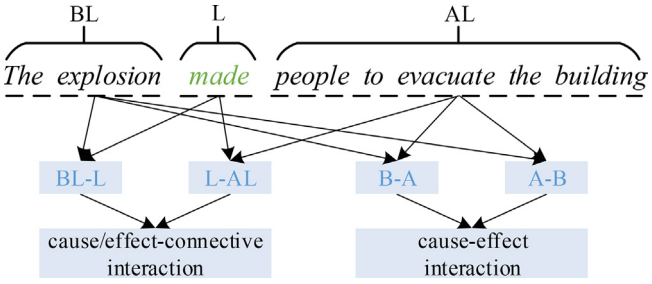

Fig. 1. An example for different segments within an sentence where "made" is the AltLex.

knowledge, such as FrameNet, WordNet, and VerbNet, to measure the correlations between words and segments, while ignoring words that never appear during the learning phase. For neural model-based methods, Oh et al. [3] propose a multi-column convolutional neural network with causality-attention (CA-MCNN) to enhance MCNNs with the causality-attention. Zhao et al. [24] claim that constructing a cause-effect network or graph could help discover co-occurrence patterns and evolution rules of causation. Therefore, Zhao et al. [2] develop causality reasoning on the heterogeneous network to extract implicit relations across sentences and discover new causal relations. To remedy the requirement of tremendous labeled training data for deep neural network models, Liu et al. [26] propose a specific reasoner to encode the background knowledge from ConceptNet, while Zuo et al. [27] employ distant supervision to extract and label data from external resources. The task objective of these two frameworks is event causality detection, also described as "relation extraction" in [28], which is different from the pipeline consisting of causality detection and cause-effect pair extraction. Finally, the current best practice [29,13] of cause-effect pair extraction is tagging the role and position of cause and effect separately. Both S-GAT [29] and SCIFI [13] utilize a self-attention mechanism to learn the token-level dependency.

In comparison to feature engineering- and neural model-based methods, our MCDN not only uses self-attention [29,13] to learn fine-grained local context and coarse-grained global long-distance dependency at the word level without any external knowledge [10,23,26]. We explicitly model causal reasoning to alleviate overweighting token-level features.

2.2. Relation network

Relation Network is initially a simple plug-and-play module to solve Visual-QA problems that fundamentally hinge on relational reasoning [30]. The original Relation Network can only perform single-step inference such as $A \rightarrow B$ rather than $A \rightarrow B \rightarrow C$. For tasks that demand complex multi-step of relational reasoning, Palm et al. [31] introduce the recurrent relational network that operates on a graph representation of objects. Relation Network can effectively couple with convolutional neural networks (CNN) [32], long short-term memory networks (LSTM) [33], and memory

networks [34] to reduce network complexity. Models obtain a general ability to reason about the relations between entities and their properties by this module. Recent progress is mainly made towards text QA and visual QA tasks. To the best of our knowledge, this is the first time that Relation Network is applied to causality detection.

3. Preliminaries

3.1. Linguistic background

This section describes the linguistic background of causal relation and the AltLex dataset that is used in the experiments. It's a widely held view that causality can be expressed explicitly and implicitly using various propositions. In the Penn Discourse Treebank (PDTB) [35], over 12% of explicit discourse connectives are marked as causal such as "hence", "as a result" and "consequently", as are nearly 26% of implicit discourse relationships. In addition, there is a class of implicit connectives in PDTB named AltLex (Alternative lexicalization) that is capable of indicating causal relations, which is an open class of markers and potentially infinite.

The definition of AltLex is extended to an open class of markers that occur within the sentence in [10]. The following are examples widespread in the new AltLexes set but are not included in PDTB explicit connectives. The word "made" with many meanings here is used to express causality. Moreover, the expression of causality in the second example is fairly ambiguous.

- Ambiguous causal verbs, e.g. *The flood* **made** *many houses to collapse.*
- Partial prepositional phrases, e.g. *They have made l4 self-driving car* **with the idea of** *a new neural network.*

According to the statistics of the parallel data constructed by Hidey and McKeown [10], there are 1144 AltLexes indicating causal relation and 7647 AltLexes indicating non-causal relation. Meanwhile, their intersection contains 144 AltLexes, which is 12.6% of causal sets and 1.8% of non-causal sets.

3.2. Notation and definitions

It is assumed that a given Wikipedia sentence S has n tokens. $S = \{s_i\}_{i=1}^{n}$ where s_i is a filtered token at position i. We use L to refer to the AltLex, BL to refer to the segment *before* the AltLex and AL to refer to the segment *after* the AltLex. Our goal is to derive a sentence-level prediction $\hat{y}$ of which the label is y as in Eq. (1). The proposed framework MCDN is shown in Fig. 3. We will detail each component in the next section.

$$y = \begin{cases} 0 & \text{sentence is non} - \text{causal} \\ 1 & \text{sentence is causal} \end{cases} \tag{1}$$

It's worth noting that Hidey and McKeown [10] utilize English Wikipedia and Simple Wikipedia sentence pairs to build a parallel corpus feature but still take one sentence as input each time. Unlike this approach, MCDN only relies on the input sentence for causal inference.

4. Methods

In this section, we elaborate on MCDN, a multi-level neural network-based method for causality detection with Transformer Encoder at the word level and SCRN at the segment level, which is primarily aimed at ambiguous and implicit relations.

4.1. Input representation

Our input representation is able to incorporate information from multi-source into a single token sequence. Inspired by [19], the representation of each token in the input sentence is constructed by summing the corresponding word, position, and segment embeddings. In contrast to BERT, the segment embeddings in this work indicate the BL, L, and AL segments in each sentence. As shown in Fig. 2, first, we adopt a word2vec toolkit [1] to pretrain word embeddings with d_{word} dimension on the English Wikipedia dump. Next, we utilize positional embeddings to map the positional information because our model has no recurrent architecture at the word level. Similarly, we use segment embeddings to incorporate more linguistic details. The dimensions of positional embeddings and segment embeddings are d_{pos} and d_{seg}, respectively. By summing the three embeddings, we obtain the vector representation $\boldsymbol{x}_i \in \mathbb{R}^d$ for token s_i where $d = d_{word} = d_{pos} = d_{seg}$. The representation $\boldsymbol{x}_i$ could provide basic features for high-level modules.

4.2. Word level

The Transformer Encoder utilized here is composed of stacked Transformer blocks. There are two sub-layers in each block: self-attention and feed-forward networks. To ensure stability and superior performance, we add a residual connection after the layer normalization for each of the sub-layers.

Self-Attention In this paper, we employ scaled multi-head self-attention, which has many advantages compared with RNN and CNN. Firstly, the "receptive field" of each token can be extended to the whole sequence without long-distance dependency diffusion. And any significant token would be assigned a high weight. Secondly, dot-product and multi-head can be optimized for parallelism separately, which is more efficient than RNN. Finally, the multi-head model aggregates information from different representation sub-spaces. For scaled self-attention, given the input matrix of n query vectors $\boldsymbol{Q} \in \mathbb{R}^{n\times d}$, keys $\boldsymbol{K} \in \mathbb{R}^{n\times d}$ and values $\boldsymbol{K} \in \mathbb{R}^{n\times d}$, computing the output attention score as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V \tag{2}$$

We take the input vector matrix $X \in \mathbb{R}^{n\times d}$ as queries, keys, and values matrix and linearly project them h times, respectively. Formally, for i-th head H_i it is formulated as below:

$$H_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

where the learned projections are matrices $W_i^Q \in \mathbb{R}^{d\times d/h}, W_i^K \in \mathbb{R}^{d\times d/h}, W_i^V \in \mathbb{R}^{d\times d/h}$. Finally, we concatenate each head and map them to the output space with $W_O \in \mathbb{R}^{d\times d}$:

$$H_O = Concat(H_1, H_2, \ldots, H_h)W^O \tag{4}$$

Feed-forward Networks We apply feed-forward networks after the multi-head self-attention sub-layer. It consists of two linear layers and a GELU activation [36] between them. Note that x is the output of the previous layer:

$$FFN(x) = GELU(xW_1 + b_1)W_2 + b_2 \tag{5}$$

where $W_1 \in \mathbb{R}^{d\times d_f}$ and $W_1 \in \mathbb{R}^{d\times d_f}$. We set $d_f = 4d$ in our experiments.

The Transformer block is stacked N times, of which the final output h_w is regarded as the representation of the sentence at the word level. We intend to handle the word with its fine-

[1] https://radimrehurek.com/gensim/

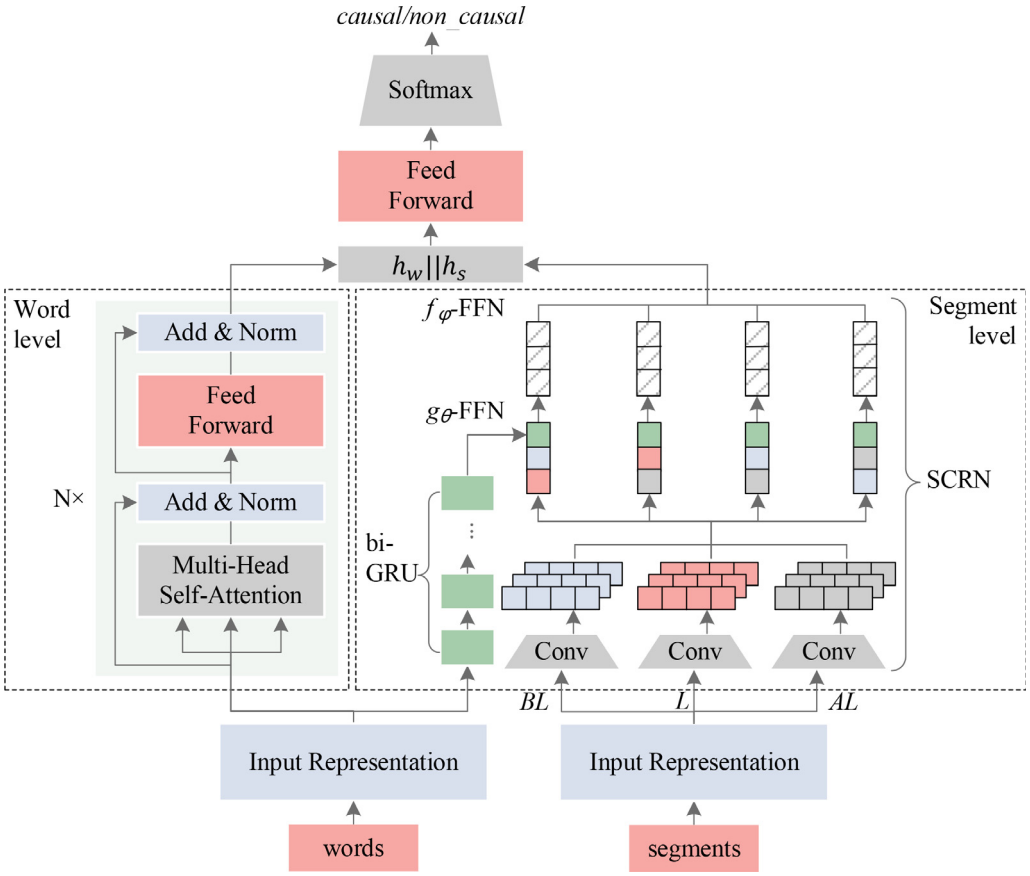

Fig. 3. The architecture of MCDN. The input sentence is split into words and segments separately and fed into the input representation layer. The left part is the word level Transformer Encoder, and the right part is the segment level SCRN.

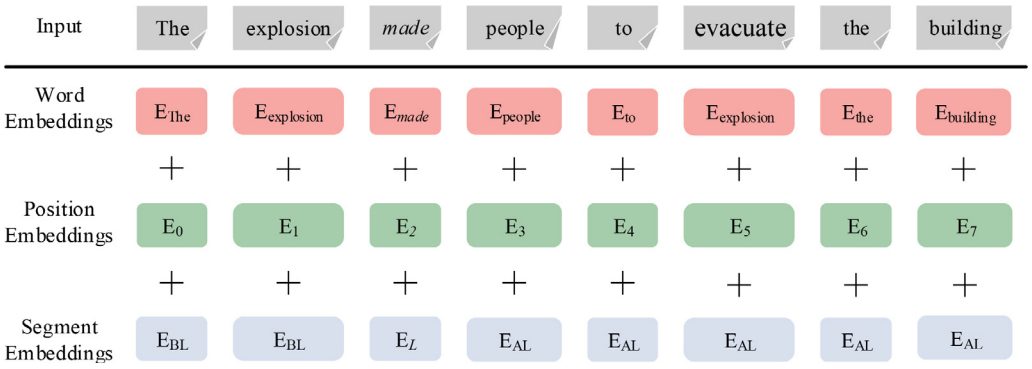

Fig. 2. MCDN input representation. The input embeddings is the sum of the word embeddings, the position embeddings, and the segmentation embeddings.

grained local context and coarse-grained global long-distance dependency information. Therefore, our word-level module could acquire not only lexico-syntax knowledge that manual patterns rarely cover but also semantic information of the words.

4.3. Segment level

We propose a novel method to infer causality within sentences at the segment level. The model is named as Self Causal Relation Network (SCRN) due to its focus on the causal relation intra-sentence compared with previous studies of Relation Network.

Dealing with segments The core idea of Relation Network is operating on *objects*. In our task, the sentence is split into three segments *BL*, *L*, and *AL* according to the position of AltLex. The input representations of these segments can be formulated as $X_{BL} \in \mathbb{R}^{T_{BL} \times d}$, $X_L \in \mathbb{R}^{T_L \times d}$, and $X_{AL} \in \mathbb{R}^{T_{AL} \times d}$ where T_{BL}, T_L, and T_{AL} are the length of tokens in each segment. Due to the variation in segment lengths, we use a three-column CNN (TC–CNN) to parse X_{BL}, X_L, and X_{AL} into a set of objects. Specifically, the representations here only employ word and segment embeddings, as the TC–CNN is capable of capturing the position information. TC–CNN convolves them through a 1D convolutional layer into k feature maps of size $T_{BL} \times 1$, $T_L \times 1$, and $T_{AL} \times 1$, where k is the sum

of kernels. The model exploits multi-scale kernels (with variable window sizes) to obtain multi-scale features. As seen in Fig. 3, the feature maps of each segment are compressed into a k-dimension vector by the max-pooling layer after convolution. Finally, we generate a set of objects for SCRN:

$$\{h_{BL}, h_L, h_{AL}\} \in \mathbb{R}^k \tag{6}$$

Dealing with the sentence The input representation X of the sentence passes through a bidirectional-GRU (bi-GRU) with d_g-dimension hidden units, and the final state $h_g \in \mathbb{R}^{2d_g}$ of the bi-GRU is concatenated to each object-pair.

SCRN We construct four object pairs concatenated with h_g. Let $\#$ denote the pair-wise operation. For causality candidates, $BL\#L$ and $L\#AL$ indicate the relation between cause-effect and AltLex, whereas $BL\#AL$ and $AL\#BL$ infer the direction of causality. The object pairs matrix $H_P \in \mathbb{R}^{4\times(2k+2d_g)}$ is shown as follows:

$$H_P = \begin{bmatrix} h_{BL\#L}; & h_g \\ h_{L\#AL}; & h_g \\ h_{BL\#AL}; & h_g \\ h_{AL\#BL}; & h_g \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} h_{BL\#L} = h_{BL}\|h_L & h_{L\#AL} = h_L\|h_{AL} \\ h_{BL\#AL} = h_{BL}\|h_{AL} & h_{AL\#BL} = h_{AL}\|h_{BL} \end{bmatrix} \tag{8}$$

Here $\|$ is a concatenation operation for the object vectors. Consequently, we modify the SCRN architecture in a mathematical formulation and obtain the final output $h_s \in \mathbb{R}^{4d_g}$ at the segment level:

$$h_s = f_\phi(\sum_i g_\theta(H_P)) \tag{9}$$

In general, the model converts the segments into object pairs using TC–CNN and passes sentences through bi-GRU to obtain the global representation. Then we combine object pairs with the global representation and make a pair-wise inference to detect the relationship among the segments. Ablation studies show that the proposed SCRN at the segment level has the capacity for relational reasoning and promotes the result significantly.

4.4. Causality detection

Our model MCDN detects the causality of each sentence based on the output h_w at the word level and h_s at the segment level. The two outputs are concatenated to provide a unified representation $h_u = h_w\|h_s \in \mathbb{R}^{d+4d_g}$. In this paper, we use a 2-layer FFN consisting of d_g units which have a *ReLU* activation followed by a *softmax* function to make the prediction:

$$FFN(h_u) = softmax(ReLU(h_u W_3 + b_3)W_4 + b_4 \tag{10}$$

In the AltLex dataset, the number of non-causal examples is over seven times the number of causal examples, resulting in an extreme example imbalance problem. If we adopt the cross-entropy (CE) loss function, the performance would be unsatisfactory. Moreover, the difficulty in detecting each example is different. For example, the sentence that contains an ambiguous AltLex such as "make" is harder to infer than that contains "cause". Consequently, we need to assign a soft weight to a causal and non-causal loss to make the model pay more attention to those examples which are difficult to detect. Motivated by Shi et al. [37], we introduce the focal loss to improve normal cross-entropy loss function. The focal loss $\mathscr{L}_{fl}$ is defined as the objective function with the balance weight hyperparameter α and the tuneable focusing hyperparameter $\beta \geqslant 0$.

$$\mathscr{L}_{fl} = \begin{cases} -\alpha(1-\hat{y})^\beta \log\hat{y} & y = 1 \\ -(1-\alpha)\hat{y}^\beta \log(1-\hat{y}) & y = 0 \end{cases} \tag{11}$$

5. Experiment

In this section, we conduct comprehensive experiments of our proposed method, MCDN. We release the code and dataset to the community for further research [2].

5.1. Datasets and evaluation metrics

Datasets We use the AltLex dataset to evaluate the proposed method. Note that in Hidey and McKeown [10], the original train set is *Training*. The *Bootstrapped* is generated using new AltLexes to identify additional examples, of which causal ones are increased by about 65 percent. In our experiment, we train all the models on the Training set and Bootstrapped set separately, use the validation set to select hyper-parameters, and then infer on the test set. Moreover, we utilize the subtask-1 dataset of SemEval-2020 Task5 [14] to validate the effectiveness of MCDN. The SemEval subtask-1 aims at recognizing counterfactual statements, e.g., "Her post-traumatic stress could have been avoided if a combination therapy had been prescribed two months earlier", and is limited to news reports from finance, politics, and healthcare domains. As the validation set is not provided by the organizer, we use 9:1 random data split for training and validation. The detailed statistical information about the datasets is listed in Table 2.

Evaluation Metrics Different evaluation metrics, including accuracy, precision, recall, and F1-score, are adapted to compare MCDN with the baseline methods. To understand our model comprehensively, we employ both Area under Receiver Operator Curve (AUROC) and Area under Precision-Recall Curve (AUPRC) to evaluate its sensitivity and specificity, particularly when causality is relatively sparse in the web text.

5.2. Implementation details

We set the initial learning rate to $1e^{-4}$ then decreased half when the F1-score has stopped increasing more than two epochs. The batch size in this experiment is 32, and the epoch size is 20. To avoid overfitting, we employ two types of regularization during training: 1) dropout for the sums of the embeddings, the outputs of each bi-GRU layer except the last, each layer in FFN and residual dropout for Transformer blocks [19]; 2) L_2 regularization for all the trainable parameters. The dropout rate is set to 0.5 and the regularization coefficient is $3e^{-4}$. In the self-attention module, we set the stack time of Transformer blocks $N = 4$ and the number of attention heads $h = 4$. In SCRN, the window sizes of TC–CNN kernels are in $[2, 3, 4]$, while the sum of kernels is $k = 150$. We use a 2-layer bi-GRU with 64 units in each direction. As for the focal loss, we set $\alpha = 0.75, \beta = 4$. For optimization, we employ Adam optimizer [38] with $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-8}$ and clip the gradients norm.

5.3. Baseline methods

In this section, the first five feature engineering-based methods are the most common class (MCC), $KLD, LS \cup KLD, LS \cup KLD \cup CONN$, and $KLD \cup LS \cup LS_{inter}$. KLD, LS, and $CONN$ represent KL-divergence score, lexical-semantic feature, and categorical feature, respectively. These methods are used as baselines in Hidey and McKeown [10]. KLD and $LS \cup KLD$ acquire the best accuracy and precision on

[2] https://github.com/shiningliang/Multi-level-Causality-Detection-Network

Table 2
Experiment Datasets Statistics.

Dataset	Type	Causal	Non-causal	Sum	Avg. Words
AltLex	*Training*	7606	79290	86896	26
	Bootstrapped	12534	88240	100744	25
	Validation	181	307	488	25
	Test	315	296	611	29
SemEval	Train	1454	11546	13000	37
	Test	738	6262	7000	38

the Training set. $LS \cup KLD \cup CONN$ and $KLD \cup LS \cup LS_{inter}$ are the best systems with the highest recall and F1-score, respectively. The following five are the most commonly used deep neural model-based methods in text classification. They are **TextCNN**, **TextRNN**, **SASE**, **DPCNN**, **DRNN**. In the experiment, we re-implement all of them. The last two baselines **DistilBERT** and **BERT** are based on pre-trained language models. For DistilBERT and BERT, we use the base-uncased version [3] and fine-tuned them on each dataset. The following are detailed descriptions of these baselines:

TextCNN [39] used here has a convolution layer, the window sizes of which are 2, 3, 4, and each has 50 kernels. Then we apply max-overtime-pooling and 2-layer FFN with ReLU activation. The dropout rate is 0.5 and $L-2$ regularization coefficient is $3e^{-4}$.

TextRNN uses a bidirectional GRU the same as the sentence encoder in SCRN and max-pooling across all GRU hidden states to obtain the sentence embedding vector, and the output layer is a 2-layer FFN. The dropout rate and L_2 regularization coefficient are the same as TextCNN.

SASE [40] employs a 2-D matrix to represent the sentence embedding, along with a self-attention mechanism and a particular regularization term. It's an effective sentence-level embedding method.

DPCNN [41] is a low-complexity word-level deep CNN model for sentiment classification and topic categorization. It can make down-sampling without increasing the number of features maps which enables the efficient representation of long-range associations.

DRNN [42] inherits the capacity to capture long-term dependencies from RNN and incorporates the position-invariance of CNN into RNN to extract key patterns. Moreover, DRNN can adjust the window size arbitrarily to different tasks.

DistilBERT [16] is a smaller language model distilled from BERT. DistillBERT has 6 Transformer layers and keeps the same hidden dimension as BERT. Nevertheless, it retains 97% of BERT performance while having 40% fewer parameters and 60% faster than BERT.

BERT [17] presented state-of-the-art results in a wide variety of NLP tasks, which is a pre-trained deep language representation model based on Transformer and Masked Language Model. BERT is inspired by transfer learning in the computer vision field, pre-training a neural network model on a known task, such as ImageNet, and then fine-tuning on a new downstream task.

It's worth noting that for the comparison on the SemEval experiments, we select the best neural model-based baseline DRNN, pre-trained language models DistilBERT and BERT. Additionally, **ON-LSTM + HAN** [43] is a publicly available work on the leaderboard with the highest score, which doesn't use any pre-trained language model in their framework.

5.4. Experiment results

We conduct each reproducible experiment 5 times and then report the average results with their standard deviation.

[3] https://github.com/huggingface/transformers

AltLex Results: Table 3 shows the results on AltLex of our model and competing methods using *Training* and *Bootstrapped* as the train set, respectively. First, we can see that MCDN dramatically outperforms all other models on both datasets by a large margin. While MCDN does not achieve the highest precision, it improves F1-score by 10% and 2% compared with the existing best feature engineering-based methods, $LS \cup KLD \cup CONN$ and $KLD \cup LS \cup LS_{inter}$. Furthermore, *KLD* feature-based SVM trained on *Training* obtains the highest precision, but has a low recall and F1-score, because it focuses on the substitutability of connectives, whereas the parallel examples usually have the same connective that would be estimated as false negatives. It is remarkable that the results of MCDN are the most robust when using *Training* and *Bootstrapped* set. Correspondingly, the feature-based linear SVM and some neural-based methods present a considerable discrepancy and obtain improvements even more than 20 on F1-score. We believe that the increase of the train set (16%) benefits the full training of these models.

Second, most neural model-based methods achieve balanced precision and recall scores except for BERT and MCDN, whose recall is much higher than precision. And only the average F1-score of MCDN is beyond 80 when trained on *Bootstrapped* set, which has the lowest standard deviation at the same time. The above results indicate that the neural model-based baselines employed here do not perform as well as MCDN. It demonstrates that causality detection is a much more complex task that requires relational reasoning capability than binary text classification, despite both can be generalized to the classification task.

SemEval Results: The results on SemEval are shown in Table 4. In the first block, when compared to the baselines DRNN and ON-LSTM + HAN, MCDN improves the F1-score by 3.80 and 3.19 on average, respectively. This verifies the generalization capability of our method. But in the second block, the performance of MCDN is slightly lower than DistilBERT and BERT. We conjecture the following reasons: i) the AltLexes set used to split segments is derived from AlteLex dataset. The number of AltLexes in this set is much larger than those of SemEval. The segment quality will be impacted by non-causal and long-tail terms; ii) The SemEval dataset is developed from new reports in three domains while the word embeddings of our model are trained on Wikipedia corpus. To some extent, the distribution of them is different; iii) The causal examples in SemEval are extremely sparse. The knowledge learned from the large pre-training corpus of BERT is superior to our method. Consequently, in the third block, we integrate DistilBERT and BERT as the input embedding of MCDN, respectively. The results show significant improvements compared with using DistilBERT or BERT alone. Moreover, we reduce the performance gap between DistilBERT and BERT after adopting our MCDN from 2.79 to 1.69. This illustrates that although the pre-trained language models provide strong word-level contextual representations, the segment-level inference capability of MCDN is still significant for causality detection.

In general, by combining the advantages of the two categories of methods, MCDN performs explicit causal reasoning based on the inherent feature from AltLex. Multi-level representation enables the model to detect causality more precisely.

Table 4
Results on SemEval.

Methods	Metrics			
	Accuracy	Precision	Recall	F1-score
DRNN	94.28 ± 0.02	78.35 ± 0.39	62.92 ± 0.92	69.79 ± 0.24
ON-LSTM + HAN	–	75.20	66.10	70.40
MCDN	74.74 ± 1.21	80.06 ± 2.10	68.25 ± 2.48	73.59 ± 0.31
DistilBERT	96.11 ± 0.12	82.10 ± 2.42	80.85 ± 2.50	81.42 ± 0.18
BERT	96.61 ± 0.15	82.84 ± 0.99	85.64 ± 0.98	84.21 ± 0.67
MCDN + DistilBERT	96.97 ± 0.06	86.94 ± 1.40	83.88 ± 1.41	85.36 ± 0.21
MCDN + BERT	97.28 ± 0.04	87.29 ± 0.63	86.81 ± 0.45	87.05 ± 0.15

6. Analysis

6.1. Ablation study of MCDN architecture

To demonstrate the synergy between different modules and their contribution to MCDN performance, we train the different modules of MCDN separately and compare their ablation comparison. The results are shown in Table 5. We can see that the full MCDN can obtain the best results most of the time. Meanwhile, SCRN plays an important role in the overall performance, which is superior to the competing baselines when trained on a relatively small dataset. It indicates the significance of causal reasoning capability for the causality detection task. While the Transformer Encoder (TE) is not strong individually, it supplies word-level semantic information complementing to segment-level inference information of SCRN and improves the performance of MCDN.

6.2. Efficiency of MCDN

Training Model with Different Data Proportion To evaluate the efficiency when the training source is limited, we adopt different proportions (0.2, 0.4, 0.6, 0.8, 1.0) of *Training* set and *Bootstrapped* set to train MCDN and several baselines. As shown in Fig. 4, both MCDN and its segment level module SCRN are not as sensitive to the proportion as the comparison methods. The performance of MCDN when the proportion is 0.2 is even better than that of others trained on the complete dataset, demonstrating that MCDN can be trained with few data and the convergence is relatively fast. In addition, DRNN and BERT obtain the largest improvement as the proportion increase from 0.2 to 1.0 on *Training* set and *Bootstrapped* set separately because they are much more difficult to train than MCDN. It's obvious that increasing the proportion is more effective in Fig. 4(b) as the density of causal examples in *Bootstrapped* set is much higher than that in *Training* set. Thus, we conjecture that it's essential to improve the performance of MCDN by building a larger dataset with high causal density.

Comparison of Model Complexity In this section, we follow the metrics in [16] and focus on the memory and time consumption reflecting the model complexity. The experiment is on a single RTX 2080Ti GPU with a batch size of 32. We report the train time and inference time for a full pass of the SemEval train set and test set. The comparison is shown in Table 6. MCDN retains 87% of BERT performance while being 4.62× smaller and 13.90× faster than BERT, which elucidates that although MCDN doesn't perform as well as the pre-trained language models, our method adequately achieves a balance between the complexity and the performance. Furthermore, as the pre-trained language models are generally applied, the results demonstrate that specific architecture like reasoning module is still effective in high-level tasks as causality detection.

Table 5
Ablation Study of MCDN Modules AltLex.

Methods	Metrics		
	F1-score	AUROC	AUPRC
Train set: AltLex *Training*			
MCDN	81.47 ± 0.61	86.37 ± 0.47	86.35 ± 0.54
- *SCRN*	62.08 ± 1.07	70.06 ± 0.18	68.93 ± 0.31
- *TE*	79.40 ± 0.67	86.08 ± 0.27	85.40 ± 0.25
Train set: AltLex *Bootstrapped*			
MCDN	82.50 ± 0.19	88.16 ± 0.10	88.61 ± 0.33
- *SCRN*	76.79 ± 0.41	80.31 ± 0.40	78.23 ± 0.79
- *TE*	82.09 ± 0.18	88.02 ± 0.19	88.75 ± 0.39

6.3. Case study

First, according to the test results, our model correctly detects part of causal relations where the AltLexes hardly appear in AltLex *Training* and *Bootstrapped* dataset, such as the example (1)(2) in Table 7. Although all the four models correctly detected the causality of the two examples, MCDN gives the largest margin of the probability, especially when the AltLex is "attribute to".

Second, the AltLex "which then" in the example (3), is an implicit clue for the causal relation between "gives rise to fibrils" and

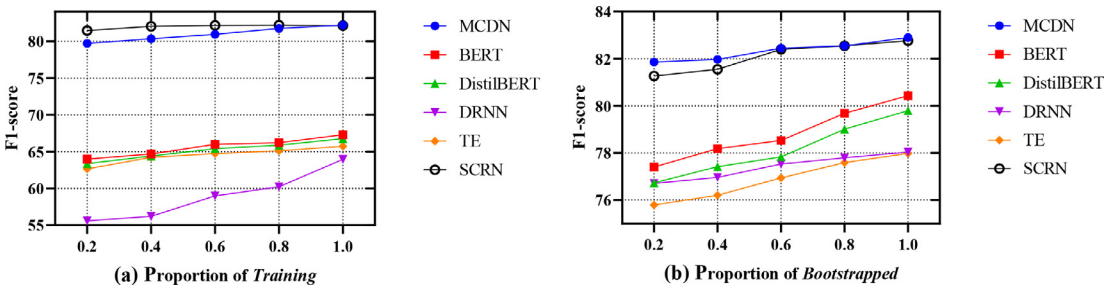

Fig. 4. Analysis of different train data proportion. We report the highest F1-score of MCDN, BERT, DRNN, Transformer Encoder, SCRN trained on *Training* set and *Bootstrapped* set after 5 runs.

Table 6
Comparison of Memory and Time Consumption. (The unit of time is seconds, and the unit of parameter is millions)

Methods	F1-score	#Tra. time (s)	#Inf. time (s)	#param. (M)
BERT	84.21(×1.00)	162.3(×1.00)	29.2(×1.00)	110(×1.00)
DistilBERT	81.42(×0.97)	61.1(×2.66)	12.4(×2.35)	66(×1.67)
DRNN	66.27(×0.79)	5.6(×28.98)	0.4(×73.00)	7.2(×15.28)
MCDN	73.59(×0.87)	26.2(×6.19)	2.1(×13.90)	23.8(×4.62)

Table 7
Predict scores of different models for the cases. (The score in **bold** represents the highest score for each case and the score in indicates it leads to a detection error.)

Labels	Examples	MCDN	BERT	SCRN	TE
(1) Causal	*The transfer was poorly received by some fans* **owing to** *a number of technical and format changes that were viewed as detrimental to the show's presentation.*	**0.999**	0.998	0.996	0.974
(2) Non-causal	*Most of the autosomal dominant familial AD can be* **attributed to** *mutations in one of three genes: those encoding amyloid precursor protein (APP) and presenilins 1 and 2.*	**0.957**	0.902	0.788	0.615
(3) Causal	*One of these fragments gives rise to fibrils of amyloid beta,* **which then** *form clumps that deposit outside neurons in dense formations known as senile plaques.*	**0.908**	0.898	0.348	0.345
(4) Non-causal	*Italy began operations in the Mediterranean, initiating a siege of Malta in June, conquering British Somaliland in August,* **making** *an incursion into British-held Egypt in September 1940.*	**0.914**	0.862	0.144	0.838

"form clumps". We find that SCRN and Transformer Encoder fail to make a correct prediction individually. However, MCDN utilizes the representation from them and obtains the highest predicted score. We conjecture that BERT has learned the commonsense knowledge about the cause-effect in the example (3) by pre-training, which contributes to detecting the causality. It's the same as in example (4) that "make" doesn't convey causality as usual here. The false prediction of SCRN is due to its characteristics. SCRN takes the segments into account while "making an incursion" is a phrase here. These two representative examples demonstrate that MCDN performs well when faced with ambiguous and implicit causality. Only the word level or the segment level information is insufficient to detect causality. MCDN can comprehend specific semantic representation in the context and infer the relations among different segments benefiting from the segment level causal reasoning module.

Finally, we investigate the misclassified examples as illustrated in Fig. 5. It reveals that "and" is the most frequent AltLex in the test set with approximately 92% prediction accuracy. The most frequent AltLexes in the false negative and false positive examples are "lead to" and "as". The accuracy for "lead to" together with its variants is 69%, slightly lower than "due to". Most false positive and false negative examples contain verbs or conjunctions as AltLex, which are not typical explicit causality connectives. In conclusion, the essential of performance improvement of MCDN is detecting ambiguous and implicit causal relations more precisely and widely.

6.4. Effectiveness for implicit causal relation detection

To verify the capability of our proposed method to detect implicit causal relations. Following Hidey and McKeown [10], apart from

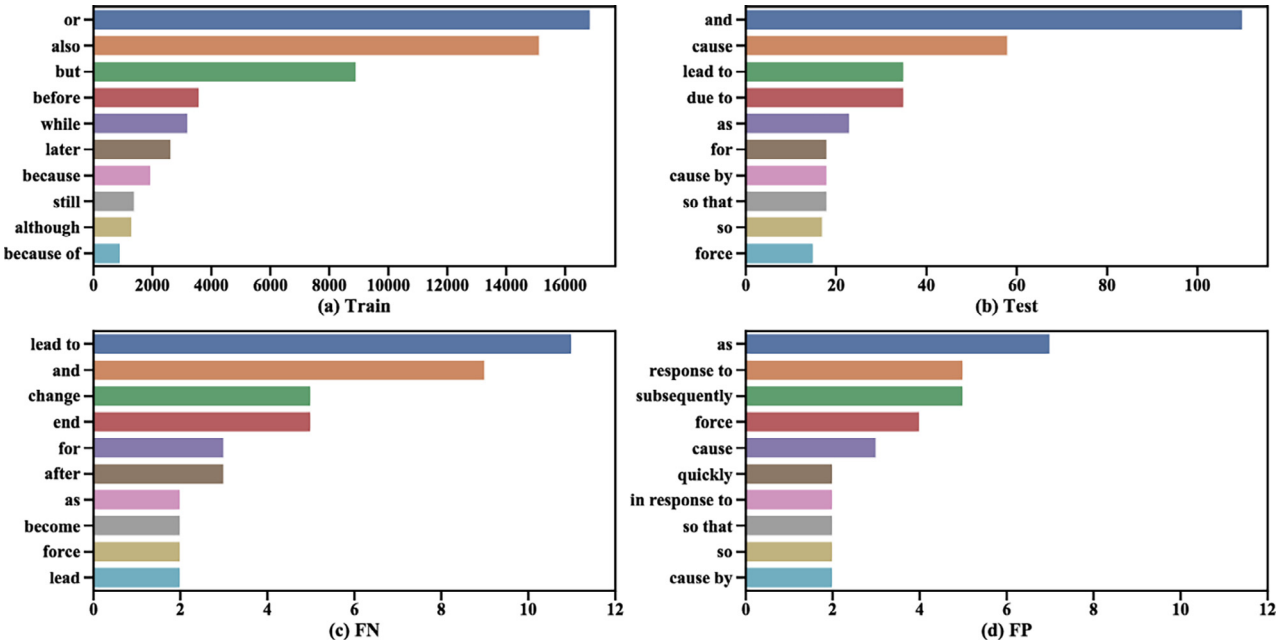

Fig. 5. Top-10 frequently appeared AltLexes in Altex *Bootstrapped* and test set. FP: false positive examples in the test result; FN: false negative examples in the test result. All the AltLexes are lemmatized.

Table 8
Comparison of implicit causal relation detection capability on the AltLex dataset. Numbers in each cell represent "True positives/Total examples of this type" in the mode predictions.

Methods	Explicit		Implicit	
	Causal	Non-Causal	Causal	Non-Causal
MCDN	31/57	141/146	234/258	92/150
BERT	37/57	117/146	224/258	97/150

Table 9
Performance of coupling with different word embeddings.

Type	F1-score	AUROC	AUPRC
word2vec (Wikipedia)	**82.50**	88.16	88.61
word2vec (Google News)	81.78	86.28	87.00
fastText	81.95	**88.36**	**89.19**
GloVe-6B	81.79	86.47	87.86
GloVe-840B	81.38	86.32	87.24
Avg.	81.88 ± 0.18	87.12 ± 0.47	87.98 ± 0.41

28 explicit causal connectives provided by the Penn Discourse TreeBank [35], we define all the rest of the connectives, i.e., conventional and newly identified AltLexes, as implicit connectives. As shown in Table 8, first respectively given the number of causal or non-causal relations examples with explicit and implicit connectives in the AltLex test set. Then we calculate the proportion of the true positives in the model predictions for the causal and non-causal relations of both connectives. The results indicate that the performance margin between MCDN and BERT is due to their implicit relation detection capability gap. Our model could mine more causal examples with implicit connectives.

6.5. Robustness of MCDN

In this section, we investigate the robustness of MCDN from two aspects. First, we alternate word representations with a different source to be word embeddings for MCDN. Then we transfer both MCDN and BERT trained on AltLex to a constructed corpus directly to demonstrate the zero-shot performance of our method.

Couple with Different Word Embeddings As is well-known, BERT uses a word-piece tokenizer to split words into sub-words, which is different from the pre-trained word embeddings used by MCDN and other models. Here we aim at evaluating the impact of different word embeddings on the performance of MCDN. Word2vec (Wikipedia) is the word embeddings used in MCDN, word2vec (Google News) [4], fastText [44], and GloVe [45] are utilized for comparison. As shown in Table 9, the discrepancy between each word embedding of the average performance is decent, which indicates the performance of MCDN is stable. The relatively weak performance of word2vec (Google News) and GloVe compared with word2vec (Wikipedia) can be attributed to there being more OOV (out-of-vocabulary) words in their dictionaries. We investigated additional methods, including trainable embedding for each OOV word and one stable embedding according to the vocabulary size for all OOV words, which result in minor performance changes.

Zero-shot Transfer In this experiment, we first filter out positive sentences containing AltLex from the "Cause-Effect" relation in the SemEval-2010-Task8 dataset and then randomly extract equivalent negative sentences from the rest relations. Finally, the corpus contains 1340 sentences, half of which contain causal relations.

To evaluate the zero-shot causality detection capability of our method, we transfer the trained MCDN to make predictions on this corpus directly. Table 10 shows that although the F1-scores drop by 5.5% and 6.6% separately, MCDN achieves significantly better results than the fine-tuned BERT, which demonstrates the transfer of our model.

[4] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit

Table 10
Zero-shot transfer results on the Constructed Corpus.

Methods	Metrics			
	Accuracy	Precision	Recall	F1-score
Train set: AltLex *Training*				
BERT	59.85	82.79	45.29	58.55
MCDN	**75.00**	**91.45**	**66.27**	**76.84**
Train set: AltLex *Bootstrapped*				
BERT	57.69	86.96	38.14	53.02
MCDN	**75.75**	**91.19**	**67.82**	**77.79**

7. Conclusion

In this paper, we propose a multi-level causality detection network (MCDN) for web text causality detection, especially implicit and ambiguous relations. To get rid of the complexity and errors of feature engineering-based methods and enhance the inference capability of neural model-based methods, we first utilize the AltLex set to split the text into segments as a prior feature without any tools to alleviate the complexity and errors of feature engineering-based methods. Then the segment set is leveraged for explicit causal reasoning at the segment level. Finally, MCDN integrates the word-level semantic information from the Transformer Encoder module and the inference information at the segment level from the SCRN module to detect the causality in the text. Extensive experimental results verify the effectiveness of our method compared with both feature engineering-based and neural network-based methods. The analysis of MCDN illustrates that MCDN makes a satisfying balance between performance and consumption, which implies the deployment potential to tackle with large-scale web text causality detection.

In the future, as a downstream task of causality detection, how to extract cause-effect pairs expressed without connectives or across the sentences is still a big challenge for the community. We also consider extending our method in this direction.

CRediT authorship contribution statement

Shining Liang: Methodology, Writing - original draft. **Wanli Zuo:** Supervision. **Zhenkun Shi:** Investigation, Conceptualization, Writing - original draft. **Sen Wang:** Writing - review & editing. **Junhu Wang:** Writing - review & editing. **Xianglin Zuo:** Revision- Experiment and Writing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is sponsored by the National Natural Science Foundation of China (61976103, 61872161), the Scientific and Technological Development Program of Jilin Province (20190302029GX, 20180101330JC, 20180101328JC), Tianjin Synthetic Biotechnology Innovation Capability Improvement Program (No. TSBICIP-CXRC-

018), and the Development and Reform Commission Program of Jilin Province (2019C053-8).

References

[1] J. Pearl, D. Mackenzie, The book of why: the new science of cause and effect, Basic books, 2018.

[2] S. Zhao, M. Jiang, M. Liu, B. Qin, T. Liu, Causaltriad: Toward pseudo causal relation discovery and hypotheses generation from medical text data, in: Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, ACM, pp. 184–193..

[3] J.-H. Oh, K. Torisawa, C. Kruengkrai, R. Iida, J. Kloetzer, Multi-column convolutional neural networks with causality-attention for why-question answering, in: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, ACM, pp. 415–424..

[4] S. Zhao, T. Liu, S. Zhao, Y. Chen, J.-Y. Nie, Event causality extraction based on connectives analysis, Neurocomputing 173 (2016) 1943–1950.

[5] S. Heindorf, Y. Scholten, H. Wachsmuth, A.-C. Ngonga Ngomo, M. Potthast, Causenet: Towards a causality graph extracted from the web, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 3023–3030..

[6] X. Ding, Z. Li, T. Liu, K. Liao, ELG: an event logic graph, arXiv preprint arXiv:1907.08015 (2019).

[7] Q.X. Do, Y.S. Chan, D. Roth, Minimally supervised event causality identification, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 294–303..

[8] P. Mirza, S. Tonelli, An analysis of causality between events and its relation to temporal information, in: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 2097–2106..

[9] T. Caselli, P. Vossen, The event StoryLine corpus: A new benchmark for causal and temporal relation extraction, in: Proceedings of the Events and Stories in the News Workshop, pp. 77–86..

[10] C. Hidey, K. McKeown, Identifying causal relations using parallel wikipedia articles, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), volume 1, pp. 1424–1433..

[11] Z. Luo, Y. Sha, K.Q. Zhu, S.-W. Hwang, Z. Wang, Commonsense causal reasoning between short texts, in: Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning, pp. 421–430..

[12] P.-W. Kao, C.-C. Chen, H.-H. Huang, H.-H. Chen, NTUNLPL at FinCausal task 2: improving causality detection using Viterbi decoder, in, in: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation, 2020, pp. 69–73.

[13] Z. Li, Q. Li, X. Zou, J. Ren, Causality extraction based on self-attentive bilstm-crf with transferred embeddings, Neurocomputing 423 (2021) 207–219.

[14] X. Yang, S. Obadinma, H. Zhao, Q. Zhang, S. Matwin, X. Zhu, Semeval-2020 task 5: Counterfactual recognition, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 322–335..

[15] D. Mariko, H. Abi-Akl, E. Labidurie, S. Durfort, H. De Mazancourt, M. El-Haj, The financial document causality detection shared task (fincausal 2020), in: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation, pp. 23–32..

[16] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019)..

[17] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2–7, 2019, Volume 1 (Long and Short Papers), pp. 4171–4186..

[18] L. Yabloko, Ethan at semeval-2020 task 5: Modelling causal reasoning in language using neuro-symbolic cloud computing, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 645–652..

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, pp. 5998–6008..

[20] R. Girju, P. Nakov, V. Nastase, S. Szpakowicz, P. Turney, D. Yuret, Semeval-2007 task 04: Classification of semantic relations between nominals, in: Proceedings of the Fourth International Workshop on Semantic Evaluations SemEval, 2007, pp. 13–18.

[21] I. Hendrickx, S.N. Kim, Z. Kozareva, P. Nakov, D. Ó. Séaghdha, S. Padó, M. Pennacchiotti, L. Romano, S. Szpakowicz, Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals, in: Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 33–38..

[22] C. Hashimoto, K. Torisawa, J. Kloetzer, M. Sano, I. Varga, J.-H. Oh, Y. Kidawara, Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), volume 1, pp. 987–997..

[23] T.N. De Silva, X. Zhibo, Z. Rui, M. Kezhi, Causal relation identification using convolutional neural networks and knowledge based features, World Academy of Science, Engineering and Technology, World Academy of Science, Eng. Technol., Int. J. Mech. Mechatronics Eng. 4 (2017) 697–702.

[24] S. Zhao, Q. Wang, S. Massung, B. Qin, T. Liu, B. Wang, C. Zhai, Constructing and embedding abstract event causality networks from text snippets, in: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, ACM, pp. 335–344..

[25] H. Kayesh, M.S. Islam, J. Wang, On event causality detection in tweets., arXiv preprint arXiv:1901.03526 (2019).

[26] J. Liu, Y. Chen, J. Zhao, Knowledge enhanced event causality identification with mention masking generalizations, in, in: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence IJCAI, 2020, pp. 3608–3614.

[27] X. Zuo, Y. Chen, K. Liu, J. Zhao, Knowdis: Knowledge enhanced data augmentation for event causality detection via distant supervision, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 1544–1550..

[28] J. Xu, W. Zuo, S. Liang, X. Zuo, A review of dataset and labeling methods for causality extraction, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 1519–1531..

[29] X. Jinghang, Z. Wanli, L. Shining, W. Ying, Causal relation extraction based on graph attention networks, J. Comput. Res. Devel. 57 (2020) 159.

[30] A. Santoro, D. Raposo, D.G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, T. Lillicrap, A simple neural network module for relational reasoning, in: Advances in neural information processing systems, pp. 4967–4976..

[31] R. Palm, U. Paquet, O. Winther, Recurrent relational networks, in: Advances in Neural Information Processing Systems, pp. 3368–3378..

[32] Y. Duan, Y. Zheng, J. Lu, J. Zhou, Q. Tian, Structural relational reasoning of point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)..

[33] W. Zheng, L. Li, Z. Zhang, Y. Huang, L. Wang, Relational network for skeleton-based action recognition, in: 2019 IEEE International Conference on Multimedia and Expo (ICME), pp. 826–831..

[34] J. Pavez, H. Allende, H. Allende-Cid, Working memory networks: Augmenting memory networks with a relational reasoning module, arXiv preprint arXiv:1805.09354 (2018)..

[35] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A.K. Joshi, B.L. Webber, The penn discourse treebank 2.0., in: LREC, Citeseer..

[36] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), arXiv preprint arXiv:1606.08415 (2016)..

[37] Y. Shi, J. Meng, J. Wang, H. Lin, Y. Li, A normalized encoder-decoder model for abstractive summarization using focal loss, in: CCF International Conference on Natural Language Processing and Chinese Computing, Springer, pp. 383–392..

[38] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014)..

[39] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751..

[40] Z. Lin, M. Feng, C.N. d. Santos, M. Yu, B. Xiang, B. Zhou, Y. Bengio, A structured self-attentive sentence embedding, arXiv preprint arXiv:1703.03130 (2017)..

[41] R. Johnson, T. Zhang, Deep pyramid convolutional neural networks for text categorization, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 562–570..

[42] B. Wang, Disconnected recurrent neural networks for text categorization, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2311–2320..

[43] X. Ou, S. Liu, H. Li, Ynu-oxz at semeval-2020 task 5: Detecting counterfactuals based on ordered neurons lstm and hierarchical attention network, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 683–689..

[44] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, Learning word vectors for 157 languages, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), pp. 3483–3487..

[45] J. Pennington, R. Socher, C.D. Manning, Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543..

Shining Liang received the B.Sc. degree from the College of Computer Science and Technology, Jilin University, Changchun, China, where he is currently pursuing the Ph.D. degree with the Key Laboratory of Symbol Computation and Knowledge Engineering, Ministry of Education. His research interests include Natural Language Processing, Causality Mining, Deep Learning and Clinical Data Mining.

Wanli Zuo is a Professor at Jilin University. He received the B.Sc. degree, M.S. degree and Ph.D. degree in the College of Computer Science and Technology, Jilin University, Changchun, China. He has published more than 160 journal papers and conference papers. His research interests include data mining, information retrieval, natural language processing, and machine learning, etc.

Zhenkun Shi received his PhD in computer science from Jilin University in 2020. He got his B.Sc degree in Software Engineering from the Agricultural University of Hebei, China. He is currently a Post-Doctoral Research Fellow in Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences. His research interests include data mining, natural language processing, and machine learning.

Sen Wang received his Ph.D. in The University of Queensland (UQ), Australia in 2014. He got his M.E. degree in Computer Science and B.Sc. degree in Computer Science from Jilin University, China and Liaoning Shihua University, China, respectively. He is a lecture in the University of Queensland. His research interests include: signal and image processing, pattern recognition and machine learning algorithms, biomedical applications and big data analytics.

Junhu Wang received his PhD in Computer Science from Griffith University, Australia in 2003. He is currently an associate professor at the School of Information and Communication Technology, Griffith University. His research interests include query processing, integrity constraint reasoning, and graph algorithms.

Xianglin Zuo received his BSc from Jilin University, China in 2015. He is now a Ph.D. candidate at College of Computer Science and Technology of the same institution, majoring in computer science. His main research interests include machine learning, deep learning, and social network analysis.