Towards Improving Causality Mining using BERT with Multi-level Feature Networks

Wajid Ali [a,1*], Wanli Zuo [a,2*], Rahman Ali [3], Gohar Rahman[4], Xianglin Zuo [a,5], and Inam Ullah[6]

[a] Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Changchun, China

[1] College of Computer Science and Technology, Jilin University, Changchun 130012, Jilin, People's Republic of China, China
[E-mail : greatforyou86@@gmail.com]

[2] College of Computer Science and Technology, Jilin University, Jilin, Changchun 130012, People's Republic of China, China
[E-mail : zuowl@jlu.edu.cn]

[3] Quaid-e-Azam College of Commerce, University of Peshawar, Peshawar 25000, Pakistan
[E-mail : rehmanali@uop.edu.pk]

[4] Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor 86400, Parit Raja, Batu Pahat, Malaysia
[E-mail : goharsa@gmail.com]

[5] College of Computer Science and Technology, Jilin University, Changchun 130012, Jilin, China
[E-mail : zuoxl17@mails.jlu.edu.cn]

[6] School of Software Engineering, Shandong University, Jinan, People's Republic of China, China
[E-mail : 2017047@mails.sdu.edu.cn]

* Correspondence: Wajid Ali, Wanli Zuo

Received March 9, 2022; revised June 18, 2022; revised June 18, 2022; revised July 25, 2022; revised September 10, 2022; accepted Septebmer 17, 2022; published October 31, 2022

Abstract

Causality mining in NLP is a significant area of interest, which benefits in many daily life applications, including decision making, business risk management, question answering, future event prediction, scenario generation, and information retrieval. Mining those causalities was a challenging and open problem for the prior non-statistical and statistical techniques using web sources that required hand-crafted linguistics patterns for feature engineering, which were subject to domain knowledge and required much human effort. Those studies overlooked implicit, ambiguous, and heterogeneous causality and focused on explicit causality mining. In contrast to statistical and non-statistical approaches, we present Bidirectional Encoder Representations from Transformers (BERT) integrated with Multi-level Feature Networks (MFN) for causality recognition, called BERT+MFN for causality recognition in noisy and informal web datasets without human-designed features. In our model, MFN consists of a three-column knowledge-oriented network (TC-KN), bi-LSTM, and Relation Network (RN) that mine causality information at the segment level. BERT captures semantic features at the word level. We perform experiments on Alternative Lexicalization (AltLexes) datasets. The experimental outcomes show that our model outperforms baseline causality and text mining techniques.

Keywords: Causality Mining, Relation Network, Multi-level Relation Network, Relation Classification, Cause-effect Relation Classification

1. Introduction

For decades, the researchers have focused on the Part-Whole, Employ-Employment Cause-Effect, Product-Producer, and If-Then relationships in the text, audio, multimedia, graphics, and video data domains. Among those data domains, the text is significant because it conveys much contextual information and preserves much human intelligence. Mining the text domain remains an inspiring job as its work with the association of semantics, sarcasm syntax, ambiguous language, and metaphors construct like figurative expressions. The basic concept and mining rules of Cause-Effect relations or causality are different among relations. Understanding causality among event pairs in natural language text (NLT) is the first and fundamental step for text understanding. Causality plays a significant role in diverse NLP applications including, decision making [1], event prediction [2], [3], generating future scenarios and medical text mining [4], [5], and question answering [6], [7] in a wide range of disciplines [8] including Computer Science [9], Environmental Sciences [10], Medicine [11], Philosophy[12], Linguistics [13], [14], and Psychology [15], [16].

In the past, non-statistical or traditional or rule-based, statistical or machine learning, and deep learning approaches are usually applied for causality mining. Non-statistical approaches are mainly based on manual work for constructing linguistic patterns (lexico-syntactic and semantic patterns) for textual feature engineering [14], [17]–[20] to mine causality in NLT. In these approaches, much human effort and time are needed for linguistics pattern engineering. Hence, no one could comprehensively achieve all linguistic patterns of causality because of its complex expression (morphological, syntactic, and lexical variations). In statistical approaches, most techniques are automated, and features are formed by refined feature engineering by using large corpora with label datasets [21]–[26], which leads to automatically mining explicit causalities, and ignoring complex and implicit causalities. In the early days, researchers designed rich syntactic, semantic, and lexical structures by polished feature engineering, where manually annotated features are wisely planned, which is used for particular domains and patterns. The performance of these approaches mainly depends on the quality of feature design, which usually depends on external NLP toolkits (Stanford Core NLP, SpaCy, AllenNLP, Apache OpenNLP) for designing. However, unfortunately, most of the toolkits are unsatisfactory and will lead to error propagation in the causality models.

In the early 80s, the idea of automatic mining information from small corpora came into consideration. Hereafter, Selfridge [27] contributed his idea to resolve difficulties in automatically mining information in a meaningful way. Inspired by his work, several researchers focused on the causality problem. The first journey toward CM was domain dependency, hand-coded linguistic features, small corpora, limited resources, and manual annotation. This journey was challenging, cost-effective, and time-consuming for researchers using diverse knowledge sources such as linguistic patterns, knowledge-based inferences, and linguistic clues. In this direction, [28] proposed the first automated causality extractor tool for acquiring knowledge from Text (TAKT) using English expository text that encodes input text into a set of propositions. This model put forward the foundation of the first 7

journey. This work has some drawbacks, encoding the input text into propositions involves extensive manual pre-processing, besides, most of the steps were domain-specific and were hard to implement for domain-independent data.

Similar to TAKT, [29] presented a PROtotype TExt Understanding System (PROTEUS), a fully functional causality extractor network of causal and temporal relations. PROTEUS is

used for equipment failure messages, named Casualty Reports (CASREPs), prepared on board ships of the U. S. Navy. This is in direct distinction with TAKT's, which uses general knowledge to learn cause-effect relations. Christopher Khoo [7], [20], [30] has published a series of influential works using linguistic clues. The possibility of their studies is restricted to the recognition of explicitly represented causality. However, they avoided domain-dependence by previous expert systems (knowledge-based) and depended on fully linguistic clues. The concerned person who reads may view a more inclusive list in Khoo's 'Ph.D.' thesis for each of these constructs [31]. Differing from [32], Girju's [6] ML paradigm for the same problem is comparatively an upfront amendment of using supervised knowledge-intensive decision tree technique C4.5 [33] to modify the semi-supervised ranking procedure and pattern authentication.

By opening benchmarked corpora for several NLP jobs, including SemEval-2007, [34] suggested task-4 by categorizing 7 often happening semantic relations among noun or noun phrases. The champions of 2 tasks [35] in SemEval-2007 task-4 and SemEval-2010 task-8 [36] use a combination of semantic, syntactic, and lexical features extracted from several NLP toolkits and knowledge bases (WordNet, FrameNet), using SVM classifier. Implicit causalities were first attempted by [22], and they replied to such queries by taking a sentence and two events occurring in the same sentence, one event can be taken as the cause of other events. The objective of this work is parallel temporal causality corpus creation. In [19], unambiguous discourse connectives are used for recognizing Alternative Lexicalizations (AltLexes) of causal relationships. They combined (FrameNet, VerbNet, and WordNet) to measure the correlations among tokens (words) and events, whereas the technique hardly grips those tokens which never been looked at in the learning phase.

In [37], causality reactions are presented for discovering adverse drug reactions (ADRs), on social media platforms (Facebook and Twitter) to automatically mine lexical patterns to represent relations among events and drugs. The purpose of this study is to notice an opposing response produced by a drug instead of just being an associated signal using causality measures. Regardless of the enhanced performance of statistical over non-statistical approaches, there are some challenges in the current systems. Firstly, most of the source corpora for causality are implicit, heterogeneous, and ambiguous, which was cost-effective and time-consuming for the prior models to extract sophisticated features of causality. Secondly, most of the features are extracted through NLP toolkits, which are error-prone and caused errors in causality mining systems. Lastly, most of the approaches are domain-specific, which needs to be re-designing for other areas. Contrary to non-statistical and statistical approaches, implementing DL techniques let the models target leftover challenges. DL techniques can automatically learn suitable features without using manual hand-crafted linguistics patterns and rules that let scholars mine distinct features with negligible domain knowledge and human effort [38].

In the past, most DL works were automated, primarily focused on explicit causalities, and overlooked implicit causalities. They detect whether the sentence or paragraph is causal or noncausal, and tiny devotion has been given to finding the direction of causal relations that which event is the cause, and which event is the effect. Though, such tasks in DL are very vital for scholars. In NLP, DL models use a discrete representation of tokens/words in vector maps called word embedding, which takes words' semantic and syntactic information [39], [40]. Pre-trained word embedding provides benefits including reduced training time and enhanced overall performance of NLP. Embedding word is a knowledgeable representation of words in a document, and the words which have identical meanings have an identical representation.

Among numerous DL models, the most widely used models for relation mining are CNN [41], RNN [42], [43], DeepCNN [44], MCNNs [45], CA-MCNN [46], FNN [47], Transformer Block [48], and BERT [49], make it possible to deal with a large number of processing tasks without complex feature engineering. Many studies had significant success in applying DL to NLP tasks including named entity recognition(NER), topic categorization(TC), sentence classification(SC), sentiment analysis(SA), and relation classification(RC). In these studies, CNN with pre-trained word embedding including, Google News[1], GloVe [50], and Pre-trained Wiki word vector[2], a distributed illustration of words in vector space [39], [40] demonstrate a significant part to encode the linguistic nature of words into a fixed-size vector to reduce the dependences on NLP toolkits [51], [52]. Some of the important approaches are discussed in the upcoming part. Silva et al. [53] applied two CNN-based techniques to identify explicit, implicit, and the direction of cause-effect relations. The problem is designed as 3 class ordering of event/entity pairs over sentence context. Class 1, identifies annotated entity pair of causal direction $e_1 \rightarrow e_2$ (Cause, Effect), Class 2, identifies $e_2 \rightarrow e_1$ (Effect, Cause), and Class 3, identifies event pairs that are non-causal.

In [45], Multi-Column CNN with background knowledge (MCNNs +BK) is presented that integrates event causality candidates and contexts with relative background knowledge from web texts, which is a variant of CNN [54]. This technique detects useful BK scattered in the web archives. It is to be noted that spreading out simple CNNs to MCNN can enhance the model's performance. Further, MCNNs, are enhanced with causality-attention-based questions and answers passage [46], called attention-Multi-column convolutional neural network (CA-MCNN) model that is not in coincidence with [45]. The attention mechanism in NN has been applied to keep a network concentrated on a specific portion of the input that appears more proper than others [55]–[57]. In [58], a novel knowledge-oriented convolutional neural network (K-CNN) is proposed for causality mining. K-CNN integrate two networks/channel, the data oriented channel (DOC) gets major causality features, and the knowledge-oriented channel (KOC) adds past human knowledge to gather the linguistic clues of cause-effect relations. In KOC, FrameNet and WordNet are used to automatically generate convolutional filters instead of training the model with a huge dataset. Besides, they used clustering, filter selection, and additional semantic features for improving the performance. In [59], the context word extension (CWE) mechanism is proposed along with Feed-forward Neural Network (FFNN), using a tweets dataset related to the 2018 commonwealth game held in Australia. They used background knowledge (BK) for event CWE, mined from news articles in causal network structure to recognize causality events. This was an exciting work because Tweets consist in an informal and unstructured format, which lacks more contextual info. In [60], a self-attentive Bi-LSTM-CRF wIth Transferred Embedding (SCITE) based approach is presented. They formulate causality as a sequence tagging problem by mining cause-effect events deprived of seeing cause-effect event pairs and their relationship. Furthermore, to enhance the performance, they used a Multi-head Self-attention [48] in the model to obtain the dependencies between causal words/tokens. First, they involved Flair embedding due to previous information deficiency [61]. Second, in text position cause and effect are rarely far away from each other. They used the SemEval-2010 task-8 extended annotated dataset. The Flair BiLSTM-CRF achieved an improvement of around 6.32% over the Bi-LSTM-CRF.

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

[2] https://fasttext.cc/docs/en/pretrained-vectors.html

In [62], a linguistically informed architecture Bi-LSTM is used for CM. They used word-level embedding and word linguistics features. This consists of 3 modules including linguistic preprocessor, resource creation, and prediction background for cause/effect events. After grouping and properly generalizing the extracted events and their relations, they created a causal graph. They used BBC News, part of SemEval2010 task-8 associated with the "Cause-Effect", and Adverse Drug Effect (ADE) dataset. The proposed model enhanced the performance more compared to state-of-the-art approaches. In [63], a Temporal causal discovery framework (TCDF) is proposed that acquires a temporal causal graph by mining causality in a time series dataset. They used multi-attention-based CNN with a causal support step. This also mines time interruption between causes and the presence of its effect. In [64], a novel deep CNN approach is presented, using only grammar tags of the nominal for mining cause-effect pairs from nominal words instead of using the exact words with their Wordnet and word2vec feature. In the past, most of the techniques used predefined syntactic and syntactic rules. On other hand, recent techniques practice shallow ML and deep neural networks on top of semantic and linguistic knowledge to categorize nominal word relations. They also used the SemEval-2010 task-8 dataset for the entire training. The drawback of this work is the overfitting issue, which is caused by a limited number of sample datasets. [65] mining causality in a short tweet text is an important and inspiring job because it contains informal characters, emojis, and other symbols. This method used a context word extension and deep causal discovery technique. In this work, 207k plus tweets associated with Commonwealth Games-2018 held in Australia are used by Twitter API. They improved the performance but has the downside of information loss.

More recent work used a head-to-tail entity annotation technique [66] that expresses the entire semantics of complex causality and clearly defines entity limits in the source sentence. They used Relation Position and Attention-graph Convolutional Networks (RPA-GCN), integrated with Graph Attention Network (GAT) and entity location perception. Where the attention layer is linked with a dependency tree to advance the network capacity to observe relational features. Additionally, a bi-directional graph convolutional network is created to further capture the deep interaction information between entities and relations. Lastly, the model iteratively predicts the relations of each word pair in the sentence and analyzes all causal pairs in the sentence by a scoring function. They used the SemEval-2010 task-8 dataset for the entire training. [67], present a generative technique for causality extraction using pointer networks and an encoder-decoder framework. They used financial domain and FinCausal for experiments and they achieved very competitive performance on this dataset. They enhance the performance compared to the state-of-the-art- technique but required much more time to produce the required result. Contrary to the reviewed statistical and non-statistical approaches, DL techniques with pre-trained word embedding are more fruitful by using automatic feature engineering techniques. Though DL models are based on a big training dataset, which covers all causality expressions in the text, a little impossible due to the diversity and ambiguity of phrases and words in the dataset. However, the ambiguous, heterogeneous, and implicit natures of causality between event pairs make them a challenging task. Though, the automatic features engineering for those causalities was hard in the existing approaches, because most of them used formal and domain-dependent datasets which contain explicit causalities. Inspired by [19], [45], [53], [58], we present a DL approach, called BERT+MFN for implicit causality recognition in the web corpus.

The prior approaches could barely incorporate the causality problem to avoid over-fitting issues, especially in the web corpus. This model combines information from the connective (AltLex) and segments (events) level of the input sentence using multi-level investigation by

parsing every word (token) with its context in the segments and connective, and recognizing causality between segments on both sides of the connective. In summary, the proposed method works as follows. We used publicly available web corpus and converted them into their specific input formats such as AB, L, and BL. After input preparation, the embedding of each word is created and added accordingly. Further, the input is passed to BERT, which deals at the word level, and MFN (Bi-LSTM, TC-KN, RN), which deals at the segment level. Lastly, the feature vector of both BERT and MFN are integrated at the last layer and passed to the classifier for causality recognition. The objectives or contributions of this article are to address and apply a DL approach. However, most of the prior studies were based on rule-based and machine-based techniques that were mostly cost-effective and had low performance. The challenges of causality mining are to train an enormous implicit, ambiguous, domain-independent, and heterogeneous dataset, which leads to causality mining as a critical task. In this article, different perspectives for causality are raised including,

- We proposed a novel deep multi-feature BERT+MFN model that tackles the causality at the tokens (words) and events (segments) levels deprived of any feature engineering.
- In BERT+MFN, BERT combines the required features at the tokens level by capturing long-range dependency and local context in the text and combining them to obtain semantic illustration at the word level that reduces limitations in feature engineering.
- In BERT+MFN, the applied novel MFN that gathers key features at the segment level, the MFN module consists of TC-KN, bi-LSTM, and RN for relational reasoning.
- The feature maps of both BERT at the token level representation (Tl_rep) are one of the shortcomings of BERT. To overcome the issue, we integrated MFN with the BERT to mine the cause-effect relationship in sentences. MFN works at segments level representation (Sl_rep) and this makes the proposed method novel. In general, the BERT+MFN has an effective reasoning potential for causality recognition.
- At the level of the event, to mine the events pairs adjacent to the Altlex (Connective), we express the sentence as segment before connective (BL) - connective (L) - segment after connective (AL). Then we might mine those segments in the formatted pairs to recognize the semantic relationship of BL – L and L – AL and Cause-Effect relationship among BL - AL / AL - BL.
- Usefulness of the proposed work is to conduct widespread experimentations in publicly accessible data. The experimental analysis presents that the proposed model outperformed baseline methods.

The remainder of this work includes. Section 2, delivers an overview of the proposed model. Section 3 presents the proposed model architecture. In Section 4, experimental analysis is discussed, while the work is concluded in Section 5.

2. BERT and Multi-level Feature Network Model

This section explores the *BERT+MFN model***,** in which BERT deals at the word level, and **MFN (**TC-KN, Bi-LSTM, and RN.**)** deals at the segment level for cause-effect relationship mining. This mainly targeted implicit, heterogeneous, and ambiguous causalities. **Fig. 1**; explore the architecture of the proposed model.

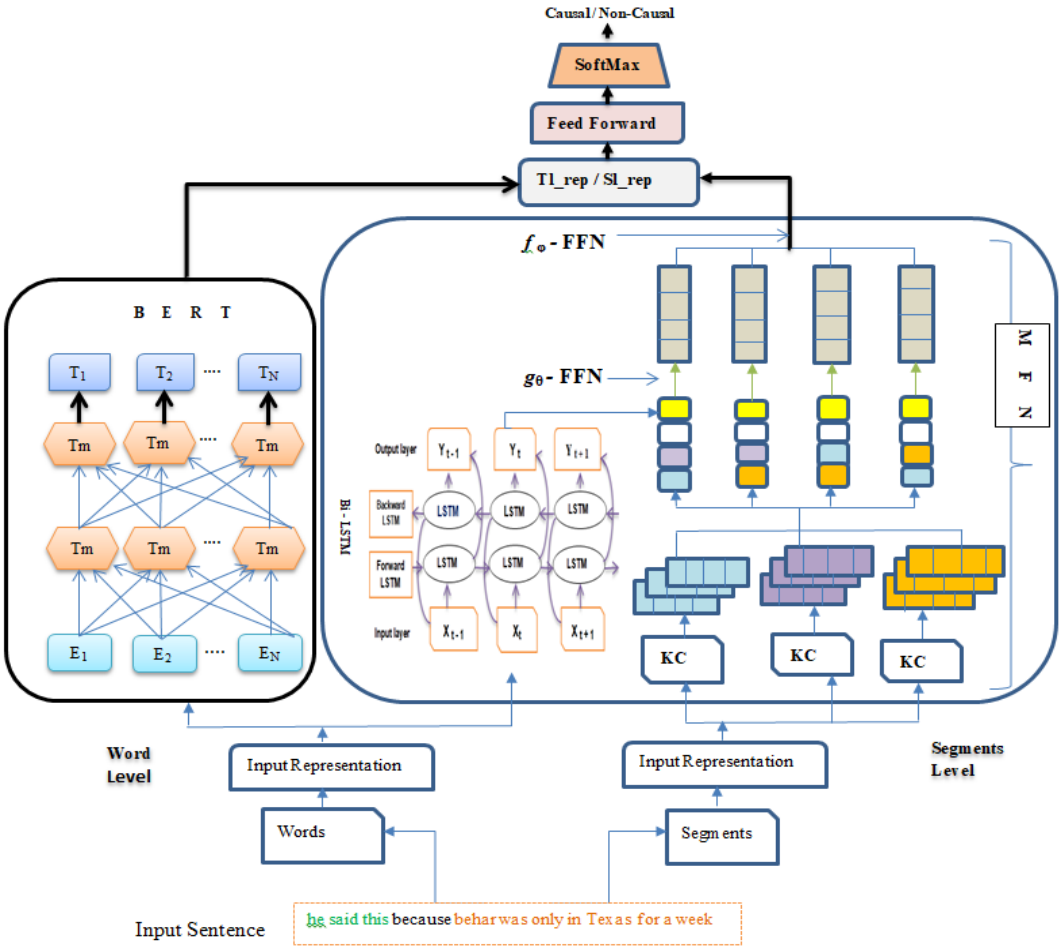


Fig. 1. The BERT+MFN architecture. The input sentence is divided into tokens/words and **segments** and given to the input representation layer. MFN work with segment level on the right side of the model and BERT works with word level on the left side of the model.

2.1 Problem Definition

For the source sentence N, it is supposed that it contains n tokens, N = {n_1, n_2... n_{i-1}, n_i}, where n_i is the filtered token at position i. Mathematically, the source sentence can be shown as a sequence of words or tokens **n** in Eq. (1).

$$N = [n_1, \quad n_2, \quad n_3, \dots, n_i] \tag{1}$$

The goal is to produce sentence level $\hat{\mathbf{y}}$ predication where **y** is the label represented by Eq. (2).

$$y = \begin{cases} 1, & Causal\ sentence \\ 0, & Non - Causal\ sentence \end{cases} \tag{2}$$

In **Fig. 2**, we used three notations to represent the target sentence at the segments and connective level, where the maximum size of AL and BL is 1-64 words and the maximum size of L is 1-8 words.

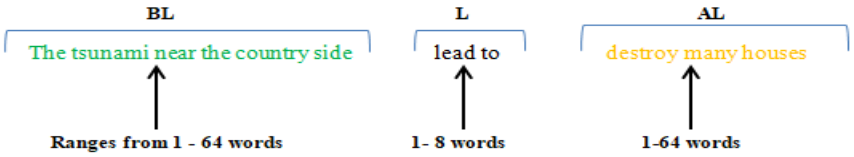


Fig. 2. Representation of input sentence at segments and connective level.

2.2 Input Representation

For input representation, we infer the effect (BL) and cause (AL) segments near Altlex (L) in the input sentence, which limits the existing challenges in RNs. These segments and Altlex are represented as (Effect (BL) - Connective (L) - Cause (AL)) input representation for causality recognition. Further, to encode those segments and connectives at token level format, the word embedding, position embedding, and connective/segment embedding of each word are combined in the source sentence [48], [49]. As shown in **Fig. 3** first, we applied a Word2Vec tool [3] to pre-train word embedding with dimension d_{wrd} in 'English Wikipedia dump' and used positional embedding of dimension $\boldsymbol{d_{pos}}$ to map token positional information, the proposed system has no recursive architecture at the word level. Likewise, in the previous works for linguistics information, our model considers segment embedding with dimension d_{seg}. Finally, summing the word, position, and segment embedding of each sentence produces a new representation $x = \{x_1, x_2, x_3 \ldots, x_{n-1}, x_n\}$ in E.q.(3), where $x_n \in R_d$ for tokens x_i in sentence N in equation (1), and $\mathrm{d} = \mathrm{d_{wrd}} = \mathrm{d_{pos}} = \mathrm{d_{seg}}$ are the dimensions of the word, position, and segment embedding that are equal. Hence, the x_i illustration makes available the basic features for high-level modules. Further, word embedding, segment embedding, and location embedding have no direct relation with each other in the sense to affect each other's features. They are just complementary information to enhance the feature engineering of each word in the input sentence.

$$X = [x_1, x_2, x_3, \ldots, x_{n-1}, x_n] \qquad (3)$$

As BERT works at the word level, so the input of the BERT is the word embedding + segment embedding (here segment embedding means which segment the word belongs to) + position embedding of each word in the sentence. In BERT, the input sentence is given in sequence order, in which each word combines its word embedding + segment embedding + position embedding. MFN works on the segment level, so the input of MFN is the addition of a word embedding + position embedding + segment embedding of word in their specific segments (AL, BL) and AltLex/Connective(L). This shows that the input embedding of the proposed model is the same but the levels of input to BERT and MFN are different.

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

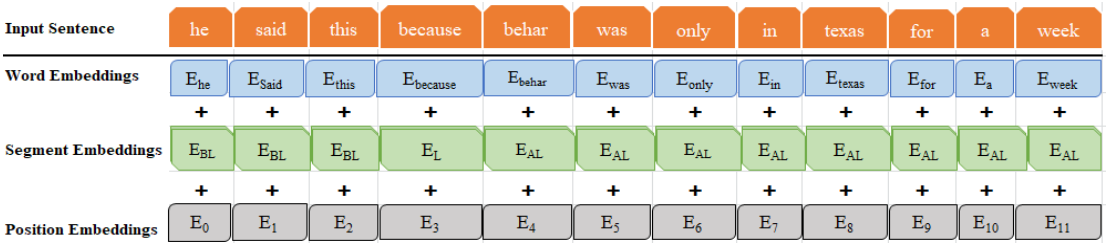


Fig. 3. BERT+MFN input illustration. Input embedding is produced by summing the word, segment, and position embedding.

2.3 Relation Network

Relation Network is a NN shape informed module for relational reasoning. The philosophy behind RN development is to constrain the functional shape of NN, which captures the key common characteristics of relational reasoning, is data-efficient, operates on a group of objects, uses versatile input format (in order invariant), and learns to infer relationships. Generally, RN in modest is a composite function, that is represented in Eq. 4.

$$RN(O) = f_{\emptyset}\left(\sum_{i,j} g_{\theta}\left(O_i, O_j\right)\right) \qquad (4)$$

The input is a set of objects (O), $O = \{o_1, o_2, o_3, \ldots, o_n\}$, $O_i \in R^m$ is the ith object, g_θ and $f_\emptyset$ are functions with parameters $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$. Here, g_θ and $f_\emptyset$ are multi-layer perceptron's (MLPs) and parameters are learnable weights, building RN end-to-end differentiable. The output of g_θ is called a relation, hence, the role of g_θ is to infer the ways that two objects are associated, or even associated at all. In Visual-Question Answering (V-QA) problems Relation Network(RN) plays a very important role, based on relational reasoning capability [68]. RN can efficiently combine with CNNs (DeepCNN, Knowledge-based CNN, and MCNN) and RNN (GRU, bi- GRU, LSTM, bi- LSTM) to increase the overall performance of causality mining models. RN usually works on object pairs for relation reasoning. The novel RN can only achieve a single-step conclusion such as, $X \rightarrow Y$ relatively $X \rightarrow Y \rightarrow Z$. For those tasks, which need difficult multistep relational reasoning, [69] introduced RNN that works on a graph illustration of objects. In [70], a complicated reasoning capability is added to memory models with RN that shortened its computational complication from nonlinear to linear. Though, their work remains only for text and V-QA. On the contrary, we enhance RN with some novel feature modules for efficient relation reasoning.

2.4 Knowledge-oriented Channels

Inspired by [58], we used a knowledge-oriented channel (KC) for efficiently recognizing cue words, cue phrases, and keywords of causality in an input sentence. In KC, we apply 'wf,' a kind of convolutional filter automatically generated from (WordNet, FrameNet) knowledge-bases, based on linguistic knowledge of causality. Compared to CNN filters, 'wf' is used to represent causality more precisely. The weights of 'wf' are the embedding of those words that are pre-trained and deprived of extra training, which would decrease the number of pre-parameters of the model considerably, which reduces the over-fitting issues in smaller datasets. **Fig. 4** represents the architecture of the KC, where the input to the KC is a sentence formatted into three segments BL, L, and AL. In this format, L is frequently used to

represent the cue phrases, cue words, and keywords (because, as result, lead to, resulted, trigger, and due to) of causality that appears among BL and AL. In a sentence, those cue phrases and keywords which appear far away from BL and AL will affect the performance of the model, as described above in Figure 2. Compared to [58], in KC, we focus on those words as an input which is included in L, BL, and AL. Similarly, to decrease the morphological dissimilarities of tokens, we used WordNet tokens to make it consistent and kept each word in its lowercase by using a lemmatizer. And each word is converted into a specific input format, described above in Figure 3. In our input format, we set the maximum size of L is 8 words, and AL and BL to 64 words. Those sentences with less than 8 and 64 words are padded by using padding characters with zero embedding vectors. The input format is created according to the maximum length of segments and connectives in the source datasets. As mentioned in **Table 1** we consider the maximum size of input sentence is about 128 for all data sets. Similarly, if we try to increase the length of the input segments and connective, then the classifier will unable to work properly because we trained the classifier with the mentioned ranges. If we increase the maximum size of the input segments and connective, then we will require a different setup of parameters and hyperparameters accordingly. In future work, we will work to deal with variant sizes of input.

2.5 Word Filters Archive Generation

We have used Algorithm 1 to automatically generate 'wf' for KC without training the network on a larger dataset. These filters are the embedding of causal words, cue words, and cue phrases, which are extracted from WordNet and FrameNet knowledge bases [71], [72]. We created many diverse size filters based on the input L, AL, and BL maximum range sizes. The maximum size of L range from 1 to 8 words, and each BL and AL range from 1 to 64 words. After, the 'wf' archive generation, the convolutional filters (c_i) for every lexical unit (lu) is formatted as $[c_1, c_2, c_3, \dots, c_i]$ in lu_k, $(k = 1, 2, 3, 4, 5, 6, 7, 8, \dots \dots 64)$, the weights of corresponding 'wf' are $f = [f_1, f_2, f_3, \dots, f_i]^T$. Where $f_i \in R_e$ is word embedding (word vector) of c_i found by seeing the word embedding table $W^{wrd} \in R^{e \times |V|}$, and k is the convolutional window sizes. We followed [58] for both 'wf' selection and clustering to make 'wf' bank generation more effective. Due to space constraints, we are unable to represent the whole procedure, but we summarized it to some extent. In step 1, we found all lexical units (lu_1-lu_{64}) of 40 causal semantic frames, recognize from FrameNet (Causation, Triggering, Reason, Response, Causation Scenario, and 34 frames start with cause), which are grouped according to the number of maximum words size (max: 1- 64).

The "lu" used in these causal frames are the remarkable clues and frequently happening words that demand causal relation in the sentence, which can be preserved as keywords, clue terms, and cue phrases for causality. In step 2, we improved these **'lu'** by considering WordNet for wide-ranging attention to causal words and automatically making a bank of causal words. In step 3, the weights of CNN's word filters originated through causal words and word embedding. Word filters created in Algorithm 1 have physical values that signify cue phrases and keywords of causality. These "wf" are created by human prior knowledge of causality that is more valuable than convolutional filters learned by training the network on a large dataset. Besides, the parameters of such filters are static values instead of free parameters in the network. Therefore, the number of preparameters in the network is pointedly reduced, which relieves the overfitting problem in a small dataset in the training. Finally, found diverse ranges of "wf" are produced from WordNet and FrameNet knowledge bases. Additionally, we have considered PropBank, VerbNet, and OntoNotes [73]–[75],

common knowledge bases to mine the linguistic knowledge of causality. VerbNet and PropBank are used to discover the reserved words of causality. Unlike FrameNet, they both are verb oriented and emphasize extra on the syntactic level. Therefore, numerous significant linguistic clues of causality including due to, lead to, because, since, and from cannot be mined, and several verbs with no causal sense may be mined unpredictably, which leads to more noises to the causality mining model. To improve the causal words originating from FrameNet, we used OntoNotes sense groupings, because word senses in 'OntoNotes' are more coarse-grained related to 'WordNet', which results in several unrelated words.

Algorithm 1: Automatic convolutional word filters generation

Step 1: Find all the lexical units of 40 causal semantic frames from FrameNet, group them according to Number of words (max: 64).

$lu_1 = \{c_1, c_2, \dots, c_{n_1}\}$

$lu_2 = \{[c_{11}, c_{12}], [c_{21}, c_{22}], \dots, [c_{n_2 1}, c_{n_2 2}]\}$

$lu_3 = \{[c_{11}, c_{12}, c_{13}], [c_{21}, c_{22}, c_{23}], \dots, [c_{n_3 1}, c_{n_3 2}, c_{n_3 3}]\}$

$lu_4 = \{[c_{11}, c_{12}, c_{13}, c_{14}], [c_{21}, c_{22}, c_{23}, c_{24}], \dots, [c_{n_3 1}, c_{n_3 2}, c_{n_3 3}, c_{n_4 4}]\}$

$lu_5 = \{[c_{11}, c_{12}, c_{13}, c_{14}, c_{15}], [c_{21}, c_{22}, c_{23}, c_{24}, c_{25}], \dots, [c_{n_3 1}, c_{n_3 2}, c_{n_3 3}, c_{n_4 4}, c_{n_5 5}]\}$

$$lu_6 = \left\{ \begin{array}{c} [c_{11}, c_{12}, c_{13}, c_{14}, c_{15}, c_{16}], [c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}], \\ [c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, c_{36}], \dots, [c_{n_6 1}, c_{n_6 2}, c_{n_6 3}, c_{n_6 4}, c_{n_6 5}, c_{n_6 6}] \end{array} \right\}$$

$$lu_7 = \left\{ \begin{array}{c} [c_{11}, c_{12}, c_{13}, c_{14}, c_{15}, c_{16}, c_{17}], [c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}, c_{27}], \\ [c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, c_{36}, c_{37}], \dots, [c_{n_7 1}, c_{n_7 2}, c_{n_7 3}, c_{n_7 4}, c_{n_7 5}, c_{n_7 6}, c_{n_7 7}] \end{array} \right\}$$

$$lu_8 = \left\{ \begin{array}{c} [c_{11}, c_{12}, c_{13}, c_{14}, c_{15}, c_{16}, c_{17}, c_{18}], [c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}, c_{27}, c_{28}], \\ [c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, c_{36}, c_{37}, c_{38}], \dots, [c_{n_8 1}, c_{n_8 2}, c_{n_8 3}, c_{n_8 4}, c_{n_8 5}, c_{n_8 6}, c_{n_8 7}, c_{n_8 8}] \end{array} \right\}$$

……………………………………………..

……………………………………………..

……………………………………………..

$$lu_{64} = \left\{ \begin{array}{c} [c_{11}, c_{12}, c_{13}, c_{14}, c_{15}, c_{16}, c_{17}, c_{18} \dots c_{164}], [c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}, c_{27}, c_{28} \dots c_{264}], \\ [c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, c_{36}, c_{37}, c_{38} \dots c_{364}], [c_{41}, c_{42}, c_{43}, c_{44}, c_{45}, c_{46}, c_{47}, c_{48} \dots c_{464}], \\ \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \\ \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \\ , [c_{n_{64} 1}, c_{n_{64} 2}, c_{n_{64} 3}, c_{n_{64} 4}, c_{n_{64} 5}, c_{n_{64} 6}, c_{n_{64} 7}, c_{n_{64} 8} \dots \dots c_{n_{64} 64}] \end{array} \right\}$$

Step 2: Extend lexical units of lu_1 by WordNet knowledge bases.

for word in lu_1 **do**

for synset in WordNet synsets of word **do**

if $\left\{ \begin{array}{c} \text{"due to", "result", "reason", "effect",} \\ \text{"cause", "}causal\text{", "}result\text{",} \\ \text{"because", "}lead\ to\text{"} \end{array} \right\}$ in WordNet gloss of

synet **then**

for lemma in WordNet lemmas of synet **do**

$\boldsymbol{if}$ $length\ of\ lemma == 1$ $\boldsymbol{then}$

$lu_1 = lu_1 + lemm$

$\boldsymbol{else\ if}$ $length\ of\ lemma == 2$ $\boldsymbol{then}$

$lu_2 = lu_2 + lemma$

$\boldsymbol{else\ if}$ $length\ of\ lemma == 3$ $\boldsymbol{then}$

$lu_3 = lu_3 + lemma$

$\boldsymbol{else\ if}$ $length\ of\ lemma == 4$ $\boldsymbol{then}$

$lu_4 = lu_4 + lemma$
else if $length\ of\ lemma == 5$ ***then***
$lu_5 = lu_5 + lemma$
else if $length\ of\ lemma == 6$ ***then***
$lu_6 = lu_6 + lemma$
else if $length\ of\ lemma == 7$ ***then***
$lu_7 = lu_7 + lemma$
else if $length\ of\ lemma == 8$ ***then***
$lu_8 = lu_8 + lemma$
…… ***then***
…….
…….. ***then***
……
else if $length\ of\ lemma == 64$ ***then***
$lu_{64} = lu_{64} + lemma$
end if
end for
end if
end for
end for

Step 3: Generate CNNs convolutional filter weights.
for each lexical unit $[c_1, c_2, c_3, \ldots, c_i]$ in $lu_k, (k = 1, 2, 3, 4, 5, 6, 7, 8, \ldots, 64)$ **do**
the corresponding filter weights is:

$$\boldsymbol{f} = [\boldsymbol{f_1}, \boldsymbol{f_2}, \boldsymbol{f_3}, \ldots, \boldsymbol{f_i}]^T$$

Where $f_i \in R^e$ is word embedding of c_i found by looking word embedding table $\boldsymbol{W^{wrd}} \in \boldsymbol{R^{e \times |V|}}$, and **k** represnet convolutional window size.
end for

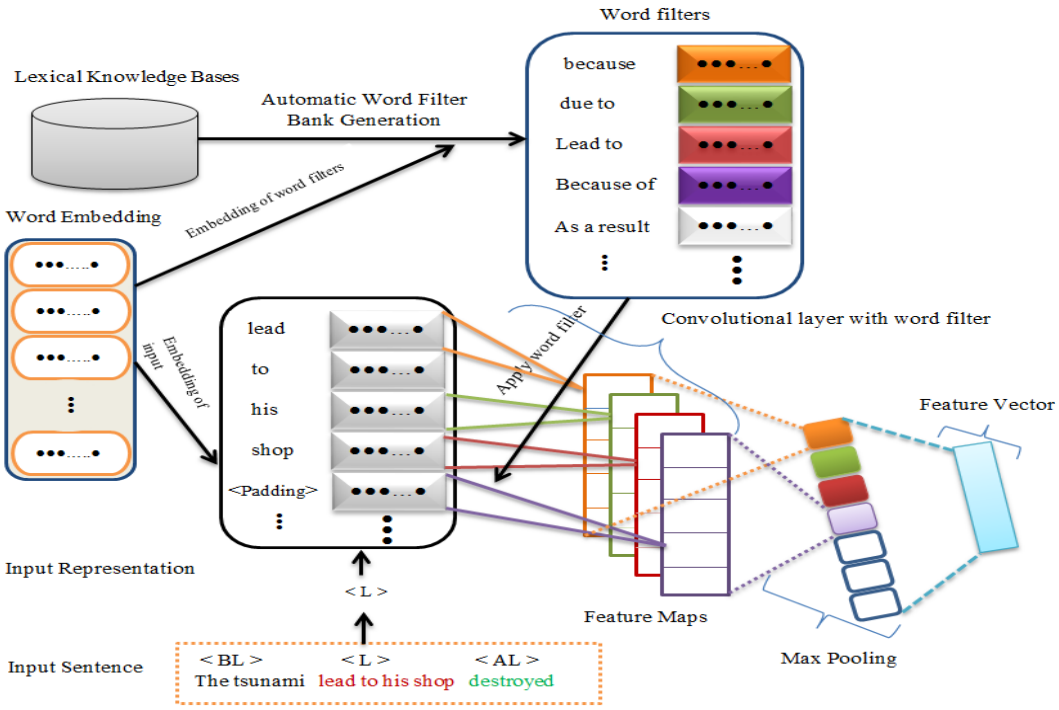


Fig. 4. Architecture of words filters archive generation

2.6 BERT for Word Level Processing

BERT depends on the original implementation of the Transformer [48]. It is very efficient to deal with word-level processing. BERT treats each word with its coarse-grained global long-distance dependency and fine-grained context-dependency information. BERT acquires lexico-syntax knowledge and lexical semantics among words, which significantly enhance the performance of the proposed model at the word level. BERT keeps sure the use of a transformer with an attention mechanism to learn contextual relationships among tokens in a text [48]. Its ordinary practice that the transformer contains two distinct mechanisms, including an encoder that reads input text and the decoder which keeps generating the prediction of the task. Since BERT proposes to produce a language system in which the encoder mechanism is only necessary. The attention mechanism of the BERT is inspired by [48]. In this section, we ignore an in-depth background description of the model architecture, hence interested readers refer to [49].

The attention layer: compared to CNN and RNN the scaled Multi-head (MH) self-attention layers have several benefits. Primarily, in the Receptive field, every token could be extended to the entire sequence, deprived of long-distance dependency distribution. A high weight would be allocated to each significant token in the sequence. Secondly, MH and dot products could be adjusted separately for parallelism that is more proficient than the transformer and RNN. Lastly, MH self-attention layer combined information from diverse subspace illustrations. In scaled attention layer (SAL), the input matrix of *n* Query vectors $Q \in R_{n \times d}$, $Keys\ K \in R_{n \times d}$, and $Values\ V \in R_{n \times d}$, which computes output attention score in Eq. (5).

$$\text{Attention}\ (Q, K, V) = \ \text{SoftMax}\left(\left.(QK)^T\right/\sqrt{(d)}\right)V \tag{5}$$

As mentioned, we took input vector-matrix $X \in R^{n \times d}$ as a Queries (Q), Keys (K), and Values (V) matrix, further linearly project them ***'h'*** times. Properly for *i-th* head, the '***Hi'*** is represented in Eq. (6).

$$H_i \ = \text{Attention}\left(QW_i^Q,\ KW_i^K,\ VW_i^V\right) \tag{6}$$

Where $W_i^Q \in R^{d \times d/h}$, $W_i^K \in R^{d \times d/h}$, and $W_i^V \in R^{d \times d/h}$ *is the* pre-trained projections matrices. Finally, every head is concatenated and map them into "MH" output space with pre-trained projection $W_{MH} \in R^{d \times d}$ in Eq. (7).

$$\text{MH} = \text{Concat}\ (H_1,\ H_2, \ldots\ldots\ldots, H_h)W_{MH} \tag{7}$$

Similarly, the same procedure is applied to the other direction because BERT is a bidirectional encoder. Due to space constraints, we are unable to mention the whole mechanism in the second direction.

Similarly, the Feed-Forward Network (FFN) layer is the second layer, which is functional next to the attention sub-layer. FFN further contains 2 sub-linear layers and ReLU inside them. It takes input x, the output of the preceding SAL layer, shown in Eq. (8).

$$FFN(x) = \text{ReLU}(xW_1 \ + b_1)W_2 + b_2 \tag{8}$$

Where W_1, $W_2 \in R^{d \times df}$ are the liner projection and $d_f = 4d$ are used in the proposed experimentation. The BERT is stacked 'N' times, where the token level representation (Tl_

rep) is the final output after passing the input sentence, which is considered as the illustration of the sentence at the tokens level.

2.7 MFN for Segment Level Processing

Proposed work used a novel technique to recognize causality in the segment level called MFN module that targets causality within a sentence compared to previous relation classification networks. MFN consists of three different units including TC-KN, bi-LSTM, and RN.

2.7.1 Three Column-Knowledge oriented Network(TC-KN) for Segments Processing

Based on earlier studies, RNs usually work on object pairs. In this unit, the input sentence is divided into three parts including L, BL, and AL, which limits the existing challenges for RNs. Further, to encode those segments and connectives at the word-level style, we integrate the word embedding, position embedding, and segment/connective embedding of each word in the source sentence [48], [49]. More, it can be represented by input formats such as $X_{BL} \in R^{T_{BL} \times d}$,$X_L \in R^{T_{L} \times d}$, and $X_{AL} \in R^{T_{AL} \times d}$. Where T_L, $T_{BL}$and T_{AL} are the length of tokens in each parts. To obtain feature maps of segments, first, we parse T_L, T_{BL}, *and* T_{AL} into a set of four pair-wise objects by using the TC-KN module of the MFN unit. This module practices linguistics information of causality at the segment level by using WordNet[4] and FrameNet[5] that capturing key linguistic signals of causality by applying convolutional word filters (wf). Where **'wf'** is a pre-trained word embedding, automatically generated from the 'Bootstraps' dataset by using Algorithm 1 in the form of a word filter archive that helps to condense the overall dimensionality of the model. TC-KN can capture the location information. Different from [76], TC-KN convolves them into one dimensional (1d) convolutional layer of ***'k'*** feature maps of size $T_{BL\times1}$, $T_{L\times1}$, and $T_{AL\times1}$ by using convolutional **'wf'** with different window sizes. Whereas, **'k'** represents the sum of kernels. After convolution, feature maps of every segment is rescaled into k-dimension features vector via the max-pooling layer. Lastly, a set of objects is produced for RN in Eq. (9).

$$\{o^{BL}, o^{L}, o^{AL}\} \in R^{k} \tag{9}$$

2.7.2 Dealing with Sentence using bi-LSTM

The bi-LSTM unit is applied at the token level to capture long-distance dependency, with this unit, the long-distance dependency and local context are combined to obtain semantic illustration at the tokens level, which decreases the limitation of feature engineering. Input representation X of the input sentence passes the bi-LSTM unit with **dg**-dimension hidden units, and the final state $\gamma \epsilon R^{2dg}$ is generated. In TC-KN, the 4 object pairs are concatenated with **'ϒ'** feature vector in Eq. (10).

[4] (https://wordnet.princeton.edu)

[5] (https://framenet.icsi.berkeley.edu/fndrupal)

$$Object\ pair = \begin{vmatrix} \gamma; & O^{BL}; & O^{L} \\ \gamma; & O^{L}; & O^{AL} \\ \gamma; & O^{BL}; & O^{AL} \\ \gamma; & O^{AL}; & O^{BL} \end{vmatrix} \quad (10)$$

Here the ";" is a concatenation symbol for the objects vector. Further, we could simplify it by using the notation in Eq. (11). Where '#' represents the pairwise action. Causality candidates *BL # L* specify the relation between cause-effect and *AltLex, and L # AL* specifies relation between AltLex and cause-effect, whereas the direction of causality is inferred by *BL # AL* and *AL # BL*.

$$O_{BL\ \#\ L} = [O_{BL};\ O_L] \qquad O_{L\ \#\ AL} = [O_L;\ O_{AL}]$$
$$O_{BL\ \#\ AL} = O_{BL};\ O_{AL} \qquad O_{AL\ \#\ BL} = [O_{AL};\ O_{BL}] \quad (11)$$

The simplified form of the resulted object pair is represented by Eq. (12).

$$\text{Object pair} = \begin{vmatrix} \gamma; & O^{BL\ \#\ L} \\ \gamma; & O^{L\ \#\ AL} \\ \gamma; & O^{BL\ \#\ AL} \\ \gamma; & O^{AL\ \#\ BL} \end{vmatrix} \quad (12)$$

So more generally, we have modified the MFN design in a mathematical formulation and gained the final segment-level representation $(Sl\_rep) \in R^{4dg}$ at the segment-level output in Eq. (13).

$$\text{Sl\_rep} = \oint \varphi(\sum g_\theta(\text{Object pair})) \quad (13)$$

At the segment-level processing, the MFN module takes TC-KN to transform all segments into object representation **'**pairs by using **KCs** and passing the target sentence to **bi-LSTM** to get the global **ϒ'**. Then integrates object pairs with **'ϒ'** feature vectors and passes to RN by making a pair-wise inference to identify the relationship between segments. In the segment level, MFN works for relational reasoning and improves the outcome considerably.

2.8 Causality Recognition

The BERT+MFN recognizes causality in the source sentence based on the output of segment level representation $\left(Sl\_rep \in R^{d+4d_g}\right)$ at the segments level and token level representation $\left(\text{Tl\_rep} \in \text{R}^{\text{d+4d}_\text{g}}\right)$ at the tokens level. Lastly, both ***Sl_rep*** and ***Tl_rep*** are integrated into a unified form in Eq. (14).

$$\text{Uni\_rep} = [Tl\_rep;\ Sl\_rep] \in R^{d+4d_g} \quad (14)$$

In our proposed model, we used 2-layer FFN that containing $d_g$units, a ReLU function, and is further followed by SoftMax for final prediction in Eq. (15).

$$\text{FFN(Uni\_rep)} = \text{SoftMax}\left(\text{ReLU}\left((Uni\_repW_1 + b_1)W_2 + b_2\right)\right) \quad (15)$$

Mass discrimination among non-causal and causal samples in the target dataset leads to a big sample inequality problem. In such a dataset, ambiguous 'AltLex' like "make" is more difficult to infer than that holds the "cause" word. Hence, it is necessary to allocate a soft weight to causal and non-causal losses to allow the classifier to pay extra attention to ambiguous samples that are hard to recognize. Inspired [77], we used the focal loss to advance the normal cross-entropy (CE) loss [78], with a tunable focusing hyper-parameter $\beta \geq 0$. The focal loss (L_{fl}) is formulated as the objective function in Eq. (16), where $\boldsymbol{\alpha}$ represents the balance weight hyper-parameter.

$$L_{fl} = \begin{cases} -a(1-\hat{y})^{\beta} \log \hat{y} & y = 1 \\ -a(1-\alpha)\hat{y}^{\beta} \log(1-\hat{y}) & y = 0 \end{cases} \quad (16)$$

We used the Adam optimizer for optimization [79] with $\beta_1 = 0.9,\ \beta_2 = 0.999,\ \in = 1e^{-8}$, and clip gradients norm.

3. Experiments

This part, explore the BERT+MFN model at the sentence level in the web corpus, which combines BERT with MFN for causality mining.

3.1 Datasets

Our model uses the AltLexes dataset, which includes the Training corpus, a total of 86896 examples, of which 7606 are causal (C) and 79290 are non-causal (NC). Bootstrapped corpus of about 100744 examples, in which 12534 of causal and 88240 are non-causal. The Bootstrapped corpus is produced utilizing new AltLexes to recognize additional from the Training corpus, which improved the causal examples. In the Dev corpus, there are 488 examples, of which 181 are causal and 307 are non-causal. Lastly, the Test corpora have 611 examples, of which 315 are causal and 296 are non-causal and fine-tunes hyperparameters on the Dev corpus. The Test set is represented by a golden annotated set. In our experiments, BERT+MFN are trained on the Bootstrapped and Training set separately, but the difference in both datasets is the number of causal examples. In the Bootstrapped dataset, the number of causal examples is little enhanced, otherwise, both the dataset are the same. **Table 1**, presents the statistics of the source dataset.

Table 1. Statistics of source corpus

Dataset	**Training Corpus**			**Bootstrapped Corpus**			**Dev Corpus**			**Test Corpus**			**Max Sentence Size (Train, Dev, Test)**
AltLexes	Total	C	NC	Total	C	NC	Total	C	NC	Total	C	NC	(128, 128,128)
	86896	7606	79290	100744	12534	88240	488	181	307	611	315	296	

3.2 Linguistics Background of Source Dataset

In this part, the linguistic background of 'AltLexes' corpus described. In the Pine Discourse Tree Bank (PDTB) [76], around 13% of explicit discourse relations are tagged causal and around 26% are implicit. Further, to those connectives, there exists another type of implicit, heterogeneous, and ambiguous relations by the name 'AltLex', which can denote causality that is unlimited and constrained as an open class of causality. In [19], the generalization of 'AltLex' was prolonged by an open class of constraints, which take place with and within sentences. Examples in the newly produced 'AltLexes' corpus do not exist in explicit relations to PDTB corpus. The below examples explore some 'Altlex' in the form of ambiguous causal verbs and partial prepositional phrases.

- Ambiguous causal verb: the accident **made** many peoples killed.
- Partial prepositional phrase: they have made reboot with **the idea of a** deep network.

The word "made" with several meanings in the first example is used to represent causality. However, in the second example, the expression of causality is rather unclear. Inspired by [19], pairs of simple English Wikipedia sentences are created for parallel corpus features and one sentence is taken as an input every time. Given statistics shows, the parallel dataset constructed in [19] has 1164 causal "AltLex", and about 7627 non-causal. In the meantime, their intersection has 155 "AltLex" that is 12.8% of causal sets and 1.9% of non-causal sets. According to statistics, the implicit, heterogeneous, and vague relations are observed in the source dataset. In such a situation, the past approaches have some demerits to make an expert model. Though, our proposed model could deal with these situations more efficiently.

3.3 Experiments Settings

Hyperparameters, initially we set the learning rate $1e^{-3}$, and then reduced it in half later of the F_1score stopped growing more than 3 epochs. We used 32 batch size and the epoch size to 20. To control the over-fitting problem, we use two kinds of regularization throughout the training including, a) Apply dropout for embedding submission, the output of each bi-LSTM layer excluding the last one, each layer in Feed-forward Network, and residual dropout of BERT blocks. b) Apply L_2 regularization to all trainable variants and set the dropout rate to 0.5, and the regularization coefficient $3e^{-4}$. In scaled attention layer, the stack time of BERT blocks is $N = 4$, and attention heads $h = 4$. In MFN, we used a total kernel $k = 150$ with various window sizes ranging from 1 to 8 filters. We apply a two-layer bi-LSTM of 64 units in every direction. For focal loss $\propto = 0.80$ and $\beta = 4.5$ are used.
Evaluation parameter, We used diverse evaluation parameters including precision, F1 score, accuracy, and recall to relate the performance of BERT+MFN with the baseline techniques. To know the proposed approach more efficiently, we used both areas under the precision-recall curve (AUPRC) and area under-receiver operator curve (AUROC) to estimate its specificity and sensitivity.

3.4 Baseline Approaches

For text classification, the most commonly used technique is TextRNN, TextCNN, DPCNN, and MCNNs. In this work these techniques are repeated, where TextRNN is based on bi-GRU, which uses max pooling through all hidden units to get sentence-level embedding vector, then applied two-layer FFN for final prediction. This procedure is identical to our proposed MFN module. TextCNN [80] has a convolution layer with filter sizes 2, 3, and 4,

where each of them has 50 kernels, max-overtime-pooling, 2-layer FFN, and ReLU activation function with 0.5 dropout rate, and $3e^{-4}$ $L2$ regularization coefficients. DPCNN [44] is a deep CNN network that works at a word level for topic categorization and sentiment classification. They do down-sampling deprived by increasing the number of feature maps, which allows proficient illustration of long-range relations. MCNNs [45] is a Multi-Column Convolutional Neural Networks that integrate event causality candidates and contexts with relative web texts (background knowledge).

3.5 Experimental Results

In **Table 2**, the detailed results of causality recognition in both Bootstrapped and Training datasets with their competing techniques are presented. We conducted every reproducible experiment in five periods and report average outcomes with standard deviation. **Table 2** represents the results of proposed and competing models on the AltLexes dataset (Training and Bootstrapped). In **Table 2**, it is reported that the proposed work on the training dataset enhances accuracy by a maximum of 21.29% and by a minimum of 17.2%, Recall by a maximum of 45.14% and minimum by 24.82%, F1-score by a maximum of 35.25% and minimum by 19.45%, and got 1.22 % low precision compared with the existing best feature engineering-based approaches. Similarly, the proposed work on Bootstrapped dataset enhances, Accuracy by a maximum of 8.77% and by a minimum of 5.82%, Recall by a maximum of 15.63% and minimum by 4.2%, F1-score by a maximum of 10.37% and minimum by 6.09%, and got 0.77 % low precision. From **Table 2**, we can notice that BERT+MFN outperforms dramatically all other models. Though BERT+MFN doesn't attain the top precision, it surges F1 score, Recall, and accuracy by competing methods. Moreover, Text CNN and feature-based SVM produces the top precision on the Training set however it achieved poor F1-score and recall since it emphasizes the substitutability of connectives through parallel samples and has similar connective that is expected false negatives.s on substitutability of connectives through parallel samples and have similar connective that is expected false negatives.

Notably, the outcomes of BERT+MFN are stronger among Bootstrapped and Training datasets. Similarly, the neural network-based techniques show a significant difference and get more enhancements. Moreover, neural network approaches tend to obtain stable precision and recall except for BERT+MFN whose recall is higher than precision. It is observed that our model outperforms all other competing models. Notably, our model is more efficient on the original Bootstrapped and Training set, whereas the neural network-based approach showed a significant difference. Our model performs well by learning distinct semantic representations of causality using BERT at the word level and MFN for causality recognition at the segment level. According to the above results, it is noted that deep convolutional approaches are stronger than knowledge-based and rule-based approaches. Additionally, BERT+MFN performed more effectively compared to the deep classification techniques that we have employed. This proves causality recognition is a difficult job that needs substantial relational reasoning ability compared to text classification.

Table 2. Analysis of causality recognition models.

Datasets		**Training Set**				**Bootstrapped Set**			
Metrics		**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Methods	Text CNN	62.36	**80.57**	43.17	53.54	76.10	76.45	73.97	76.14
	Text RNN	61.87	76.62	37.46	50.32	75.61	75.78	77.46	76.61
	DP CNN	64.97	69.06	58.10	63.10	77.09	**79.66**	74.60	77.05
	MCNNs	66.45	68.97	63.49	66.12	78.56	75.99	85.40	80.42
	BERT+MFN	**83.65**	79.35	**88.31**	**85.57**	**84.38**	78.89	**89.60**	**86.51**

For more informative analysis of the results, we have drawn **Fig. 5(a)**, for training dataset, and **Fig. 5(b)** for bootstrapped dataset.

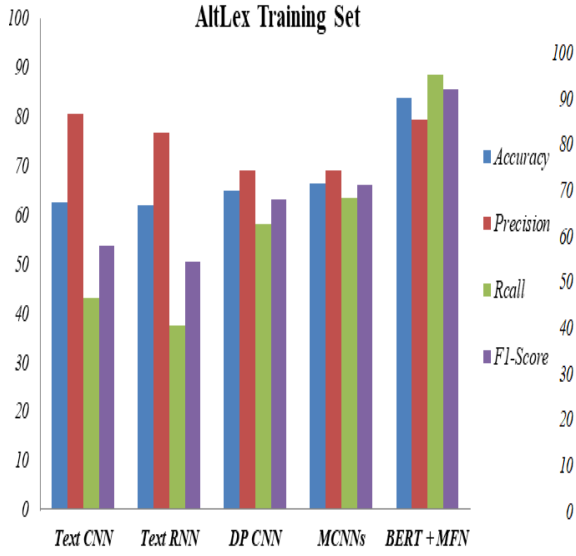


Fig. 5(a). Analysis of proposed model using Training set with state-of-the-art techniques.

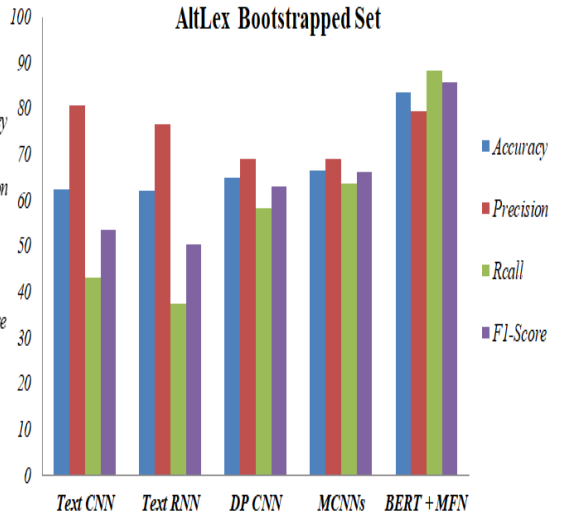


Fig. 5(b). Analysis of proposed model using Bootstrapped set with State-of-the-art techniques.

3.6 Analyses

3.6.1 Ablation Analysis of Proposed Work

It is imperative to explain and explore each part of the BERT+MFN along with their contribution. We conducted an ablation comparison by training different parts of the model separately. **Table 3**, represents the results set on both datasets, which shows that BERT+MFN obtains significant F1-score, AUROC, and AUPRC scores on training AltLex, and similarly, on bootstrapped AltLex it obtains significant F1-score and AUROC. In our

model both, BERT and MFN played a key role, in which MFN demonstrates the significance of relational reasoning capacity. BERT demonstrates the significance of the model at the word level. Though individually, they both are not very powerful for mining causality, and however, they provide important complementary illustrations at the word and segments level, which improves the overall performance of our model.

Table 3. Ablation Analysis

Methods	***Metrics***		
Train Dataset: Training AltLex	**F1-Score**	**AUROC**	**AUPRC**
BERT+MFN	**85.57**	**87.44**	**88.38**
MFN	65.11	68.91	69.10
BERT	79.17	87.19	87.30
Train Dataset: Bootstrapped AltLex	**F1-Score**	**AUROC**	**AUPRC**
BERT+MFN	**86.51**	**85.34**	84.33
MFN	79.93	79.62	75.53
BERT	*86.26*	85.20	**84.49**

As shown in **Fig. 6 (a), (b)**, the Ablation analysis of BERT+MFN, MFN, and BERT on Train Dataset: Training AltLex and Train Dataset: Bootstrapped AltLex, which make more sense for the reader.

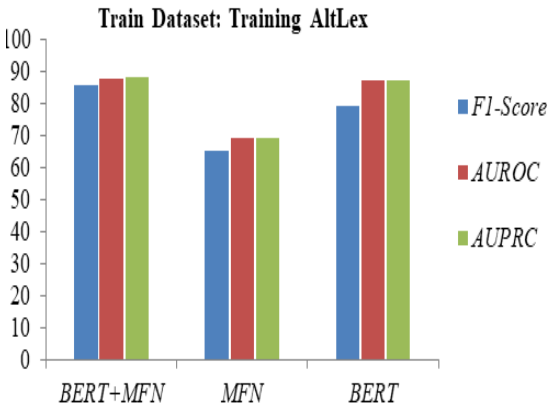


Fig. 6(a). Ablation analysis on Training AltLex.

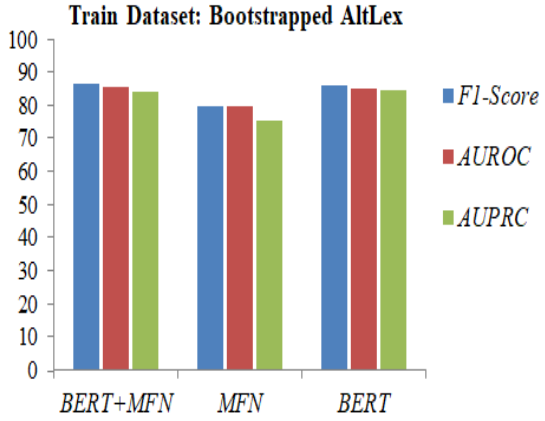


Fig. 6(b). Ablation analysis on Bootstraped AltLex.

4. Conclusion and Future direction

This work recognizes implicit, heterogeneous, and ambiguous causality in the web corpus using a multi-level causality recognition model, called the BERT+MFN model. BERT deals with web sentences at the word levels and MFN (bi-LSTM, TC-KN, and RN) deals with sentences at the segments level by combing information from data using bi-LSTM at the

token level to capture long-distance dependency, with this unit, long-distance dependency and local context are combined to obtain semantic illustration at the tokens level, which decreases the limitation of feature engineering, with TC-KN human prior knowledge are extracted from lexical knowledge bases to build an integrated sentence for causality recognition. We used convolutional 'wf' in TC-KN, which is automatically produced by linguistic knowledge of causality in WordNet and FrameNet knowledge archives. These 'wf' denote important linguistic clues of causality, letting the model mine these linguistic clues more precisely. Where NN is a relational reasoning module that captures the key common characteristics of relational reasoning. Compared to different causality and text mining approaches, our model played a notable role that inference complicated causality at the sentence level. Some challenges in the current system still exist, including implicit causality through sentences, implicit entities pairs recognition within sentences, and implicit entities pairs recognition through sentences. Hence, in the future, it is necessary to consider these challenges by creating a model to deal with implicit causality through sentences, to deal with implicit entity pairs within the sentence, and deal with implicit causality through sentences. Further, more enhanced features and external background knowledge are needed to strengthen the future direction.

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, and 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] A. Miranda and E. Jacobo, "Extracting a causal network of news topics," in *Proc. of Move to Meaningful Internet Syst. OTM 2012 Work*, Rome, Italy, pp. 33–42, 2012.

[2] K. Radinsky, S. Davidovich, and S. Markovitch, "Learning causality for news events prediction," in *Proc. of 21st international conference on World Wide Web, Lyon France*, pp. 909–918, 2012. Article (CrossRef Link).

[3] C. Silverstein, S. Brin, R. Motwani, and J. Ullman, "Scalable techniques for mining causal structures," *Data Min. Knowl. Discov.*, vol. 4, pp. 163–192, 2000. Article (CrossRef Link).

[4] M. Riaz and R. Girju, "Another Look at Causality: Discovering Scenario-Specific Contingency Relationships with No Supervision," in *Proc. of 2010 IEEE Fourth International Conference on Semantic Computing, Pittsburgh*, PA, USA, pp. 361–368, 2010. Article (CrossRef Link).

[5] C. Hashimoto et al., "Toward Future Scenario Generation: Extracting Event Causality Exploiting Semantic Relation, Context, and Association Features," in *Proc. of 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, Maryland, Vol. 1, pp. 987–997, 2014. Article (CrossRef Link).

[6] R. Girju, "Automatic Detection of Causal Relations for Question Answering," in *Proc. of ACL 2003 workshop on Multilingual summarization and question answering*, Sapporo, Japan, Vol 12, pp. 76–83, 2003. Article (CrossRef Link).

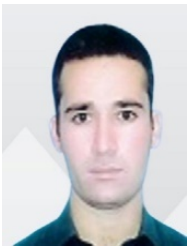
[7] C. Khoo, J. Kornfilt, O. RN, and S. MYAENG, "Automatic extraction of cause-effect information from newspaper text without knowledge-based inferencing," *Lit. Linguist. Comput.*, Vol. 13(4), pp. 177–186, 1998. Article (CrossRef Link).

[8] K. Chan and W. Lam, "Extracting causation knowledge from natural language texts," *Int. J. Intell. Syst.*, Vol. 20(3), pp. 327–358, Mar 2005. Article (CrossRef Link).
[9] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal protein-signaling networks derived from multiparameter single-cell data," *Sci.*, Vol. 308(5721), pp. 523-529, Apr.2005. Article (CrossRef Link).
[10] P. L. Araúz and P. Faber, "Causality in the Specialized Domain of the Environment," *Semantic relations-II. Enhancing resources and applications workshop programme, Lütfi Kirdar Istanbul Exhibition and Congress Centre*, Turkey, pp. 10, 2012.
[11] C. Khoo, S. Chan, and Y. N., "Extracting causal knowledge from a medical database using graphical patterns," in *Proc. of the 38th Annual Meeting on Association for Computational Linguistics*, pp. 336–343, 2000. Article (CrossRef Link).
[12] White Peter A., "Ideas about causation in philosophy and psychology," *Psychol. Bull.*, Vol.108, no.1, pp. 3-18, 1990. Article (CrossRef Link).
[13] L. Talmy, *Toward a cognitive semantics. Concept structuring systems*, Vol.1, Conceptual structuring systems, Cambridge, MA: MIT press, 2000, pp. 1–565.
[14] J. R. Hobbs, "Toward a Useful Concept of Causality for Lexical Semantics," *J. Semant*, Vol. 22(2), pp. 181–209, 2005. Article (CrossRef Link).
[15] Wolff Phillip and Song Grace., "Models of causation and the semantics of causal verbs," *Cogn. Psychol.*, vol. 47, no. 3, pp. 276–332, 2003. Article (CrossRef Link).
[16] Wolff Phillip, "Representing causation," *J. Exp. Psychol.*, vol. 136(1), pp. 82-111, 2007. Article (CrossRef Link).
[17] H. C. Bozsahin and N. V. Findler, "Memory-Based Hypothesis Formation: Heuristic Learning of Commonsense Causal Relations from Text," *Cogn. Sci.*, vol. 16, no. 4, pp. 431–454, 1992. Article (CrossRef Link).
[18] A. Hashimy, S. H. Amaal, and K. Narayanan, "Ontology enrichment with causation relations," in *Proc. of 2013 IEEE Conference on Systems, Process & Control (ICSPC)*, IEEE, pp. 186–192, 2013. Article (CrossRef Link).
[19] C. Hidey and K. Mckeown, "Identifying Causal Relations Using Parallel Wikipedia Articles," in *Proc. of 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,Berlin, Germany, pp. 1424–1433, 2016. Article (CrossRef Link).
[20] C. Khoo and S. Chan, "Extracting causal knowledge from a medical database using graphical patterns," in *Proc. of 38th annual meeting of the association for computational linguistics*, Hong Kong, pp. 336–343, 2000. Article (CrossRef Link).
[21] N. Asghar, "Automatic Extraction of Causal Relations from Natural Language Texts: A Comprehensive Survey," pp.1-10, May 2016. Article (CrossRef Link).
[22] S. Bethard and J. H. Martin, "Learning Semantic Links from a Corpus of Parallel Temporal and Causal Relations," in *Proc. of 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, Columbus Ohio, pp. 177-180, 2008. Article (CrossRef Link).
[23] Z. Luo, Y. Sha, K. Q. Zhu, and Z. Wang, "Commonsense Causal Reasoning between Short Texts," in *Proc. of Fifteenth International Conference on Principles of Knowledge Representation and Reasoning, KR'16, Cape Town*, South Africa, pp. 421–430, 2016.
[24] P. Pakray and A. Gelbukh, "An open-domain cause-effect relation detection from paired nominals," in *Proc. of Mexican International Conference on Artificial Intelligence*, Springer, Cham, pp. 263–271, 2014. Article (CrossRef Link).
[25] S. Sasaki, S. Takase, N. Inoue, N. Okazaki, and K. Inui, "Handling Multiword Expressions in Causality Estimation," in *Proc. of IWCS 2017-12th International Conference on Computational Semantics-Short papers*, pp. 1–6, 2017.
[26] X. Yang and K. Mao, "Multi level causal relation identification using extended features," *Expert Syst. Appl.*, vol. 41, no. 16, pp.7171-7181, 2014. Article (CrossRef Link).
[27] M. Selfridge, "Toward a natural language-based causal model acquisition system," *Appl. Artif. Intell.*, Vol.3, pp. 191–212, 1989. Article (CrossRef Link).

[28] K. Randy M and B.-R. Genevieve, "Knowledge-based acquisition of causal relationships in text," *Knowl. Acquis.*, Vol. 3(3), pp. 317–337, 1991. Article (CrossRef Link).
[29] R. Grishman, "Domain Modeling for Language Analysis," *Linguist. Approaches to Artif. Intell.*, pp. 1-8, 1988.
[30] C. Khoo, S. Chan, Y. Niu, and A. Ang, "A method for extracting causal knowledge from textual databases," *Singapore J. Libr. Inf. Manag.*, Vol.28, pp. 48–63, 1999. Article (CrossRef Link).
[31] C. Khoo, "Automatic identification of causal relations in text and their use for improving precision in information retrieval," 1995. Article (CrossRef Link).
[32] R. C. Schank, *Dynamic memory : a theory of reminding and learning in computers and people*, Cambridge University Press:New York, NY, United States, 1983, pp. 234.
[33] J. Quinlan, "C4.5:programs for machine learning," *Mach. Learn*, Vol.16, pp. 235–240, 1994.
[34] R. Girju, P. Nakov, V. Nastase, S. Szpakowicz, P. Turney, and D. Yuret, "Classification of semantic relations between nominals," *Lang. Resour. Eval.*, vol. 43, no. 2, pp. 105-121, Jun. 2009. Article (CrossRef Link).
[35] R. Girju, B. Beamer, and A. Rozovskaya, "A knowledge-rich approach to identifying semantic relations between nominals," *Inf. Process. Manag.*, Vol. 46(5), pp. 589–610, 2010. Article (CrossRef Link).
[36] B. Rink and S. Harabagiu, "UTD: Classifying Semantic Relations by Combining Lexical and Semantic Resources," in *Proc. of 5th international workshop on semantic evaluation*, Uppsala University, Uppsala, Sweden, pp. 256–259, 2010.
[37] D. Bollegala and S. Maskell, "Causality patterns for detecting adverse drug reactions from social media: text mining approach," *JMIR public Heal. surveillance*, Vol.4(2), pp. e8214, 2018, Article (CrossRef Link).
[38] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, "Deep learning applications and challenges in big data analytics," *J. big data*, Vol. 2(1), pp. 1–21, 2015. Article (CrossRef Link).
[39] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in *Proc. of ICLR Workshops Track*, pp. 1–12, 2013.
[40] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: A simple and general method for semi-supervised learning," in *Proc. of 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, pp. 384–394, 2010.
[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017. Article (CrossRef Link).
[42] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," in *Proc. of ICASSP 2015 - 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia*, pp. 4520-4524, 2015. Article (CrossRef Link).
[43] K. Cho et al., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv Prepr. arXiv1406.1078.*, pp. 1–15, 2014.
[44] R. Johnson and T. Zhang, "Deep Pyramid Convolutional Neural Networks for Text Categorization," in *Proc. of 55th Annual Meeting of the Association for Computational Linguistics*, Vol.1, Long Papers,Vancouver, Canada, pp. 562–570, 2017. Article (CrossRef Link).
[45] C. Kruengkrai, K. Torisawa, C. Hashimoto, J. Kloetzer, J.-H. Oh, and M. Tanaka, "Improving Event Causality Recognition with Multiple Background Knowledge Sources Using Multi-Column Convolutional Neural Networks," in *Proc. of AAAI Conference on Artificial Intelligence*, San Francisco, California, USA, Vol. 31, No. 1, pp. 3466–3473, 2017. Article (CrossRef Link).
[46] J. Oh, K. Torisawa, C. Kruengkrai, and I. R, "Multi-column convolutional neural networks with causality-attention for why-question answering," in *Proc. of Tenth ACM International Conference on Web Search and Data Mining*, Cambridge, pp. 415–424, 2017. Article (CrossRef Link).

[47] E. M. Ponti and A. Korhonen, "Event-related features in feedforward neural networks contribute to identifying causal relations in discourse," in *Proc. of 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, Valencia, Spain, pp. 25-30, 2017. Article (CrossRef Link).
[48] A. Vaswani et al., "Attention Is All You Need," in *Proc. of 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, pp. 5998–6008, 2017. Article (CrossRef Link).
[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. of 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT, Minneapolis, MN, USA,Vol. 2, pp. 4171–4186, 2019. Article (CrossRef Link).
[50] J. Pennington, R. Socher, and C. M.-P., "Glove: Global vectors for word representation," in *Proc. of 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014. Article (CrossRef Link).
[51] T. H. Nguyen and R. Grishman, "Relation Extraction: Perspective from Convolutional Neural Networks," in *Proc. of NAACL-HLT 2015*, Denver, Colorado, pp. 39-48, 2015. Article (CrossRef Link).
[52] C. N. Dos Santos, B. Xiang, and B. Zhou, "Classifying relations by ranking with Convolutional neural networks," in *Proc. of ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*, Vol.1, pp. 626-634, 2015. Article (CrossRef Link).
[53] T. De Silva, X. Zhibo, and R. Z, "Causal relation identification using convolutional neural networks and knowledge based features," *Int. J. Comput. Syst. Eng.*, pp. 697–702, 2017. Article (CrossRef Link).
[54] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," in *Proc. of 2012 IEEE conference on computer vision and pattern recognition*, Providence, Rhode Island, pp. 3642-3649, 2012. Article (CrossRef Link).
[55] J. L. Ba, V. Mnih, and K. Kavukcuoglu, "Multiple object recognition with visual attention," in *Proc. of 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015. Article (CrossRef Link).
[56] J. Chorowski and D. Bahdanau, "Attention-Based Models for Speech Recognition," 2015. Article (CrossRef Link).
[57] A. M. Rush, S. Chopra, and J. Weston, "A Neural Attention Model for Abstractive Sentence Summarization," in *Proc. of EMNLP 2015*, 2015. Article (CrossRef Link).
[58] P. Li and K. Mao, "Knowledge-oriented Convolutional Neural Network for Causal Relation Extraction from Natural Language Texts," *Expert Syst. with Appl.*, vol. 115, pp. 512–523, 2019. Article (CrossRef Link).
[59] H. Kayesh, M. S. Islam, and J. Wang, "On Event Causality Detection in Tweets," *arXiv Prepr. arXiv1901.03526*, pp. 1–8, Jan. 2019. Article (CrossRef Link).
[60] Z. Li, Q. Li, X. Zou, and J. Ren, "Causality extraction based on self-attentive BiLSTM-CRF with transferred embeddings," *Neurocomputing*, Vol. 423, pp. 207-219, Jan. 2021. Article (CrossRef Link).
[61] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual String Embeddings for Sequence Labeling," in *Proc. of 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA, pp. 1638-1649, 2018.
[62] T. Dasgupta, R. Saha, L. Dey, and A. Naskar, "Automatic extraction of causal relations from text using linguistically informed deep neural networks," in *Proc. of 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia*, pp. 306-316, 2018. Article (CrossRef Link).
[63] M. Nauta, D. Bucur, and C. Seifert, "Causal Discovery with Attention-Based Convolutional Neural Networks," *Machine Learning and Knowledge Extraction*, Vol. 1(1), Jan., pp. 312–340, 2019. Article (CrossRef Link).

[64] R. Ayyanar, G. Koomullil, and H. Ramasangu, "Causal Relation Classification using Convolutional Neural Networks and Grammar Tags," in *Proc. of 2019 IEEE 16th India Council International Conference (INDICON), Marwadi University, Rajkot (GUJARAT)*, India, pp. 1-3, 2019. Article (CrossRef Link).
[65] H. Kayesh, M. S. Islam, J. Wang, A. S. M. Kayes, and P. A. Watters, "A deep learning model for mining and detecting causally related events in tweets," *Concurr. Comput. Pract. Exp.*, vol. 34, no. 2, pp. 1-15, 2022. Article (CrossRef Link).
[66] Vivek, Md Imbesat Rizvi, Jessica Huber, Paige Bartusiak, Bogdan Sacaleanu, and A. F. Khetan, "MIMICause: Representation and automatic extraction of causal relation types from clinical notes," *Findings of the Association for Computational Linguis*, pp. 764-773, 2022. Article (CrossRef Link).
[67] T. Nayak, S. Sharma, Y. Butala, K. Dasgupta, P. Goyal, and N. Ganguly, "A Generative Approach for Financial Causality Extraction," in *Proc. of the Web Conference 2022 (WWW '22 Companion)*, Virtual Event, Lyon, France, pp. 579-578, 2022. Article (CrossRef Link).
[68] A. Santoro et al., "A simple neural network module for relational reasoning," *Advances in neural information processing systems*, pp. 1–16, 2017. Article (CrossRef Link).
[69] R. B. Palm, U. P. Deepmind, and O. Winther, "Recurrent Relational Networks," in *Proc. of 32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, Montréal, Canada, pp. 1–11, 2018. Article (CrossRef Link).
[70] J. Pavez, H. Allende, F. S. María, and H. Allende-Cid, "Working Memory Networks: Augmenting Memory Networks with a Relational Reasoning Module," in *Proc. of ACL 2018*, pp. 1-10, 2018. Article (CrossRef Link).
[71] R. Poli, M. Healy, and A. Kameas, *Theory and applications of ontology: Computer applications*, 2010. Article (CrossRef Link).
[72] J. Ruppenhofer, M. Ellsworth, M. R. L. Petruck, C. R. Johnson, and J. Scheffczyk, "FrameNet II: Extended theory and practice. International Computer Science Institute," pp. 1–119, 2016.
[73] P. Kingsbury and M. Palmer, "PropBank: the Next Level of TreeBank," *Treebanks and lexical Theories*, vol. 3, pp. 1–12, 2003.
[74] K. Schuler, "VerbNet: A broad-coverage, comprehensive verb lexicon," *University of Pennsylvania*, 2005.
[75] E. Hovy, M. Marcus, M. Palmer, and L. Ramshaw, "OntoNotes: the 90% solution," in *Proc. of human language technology conference of the NAACL*, Companion Volume: Short Papers, pp. 57–60, 2006.
[76] R. Prasad et al., "The Penn Discourse TreeBank 2.0," in *Proc. of Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, pp. 1–8, 2008.
[77] Y. Shi, J. Meng, J. Wang, H. Lin, and Y. Li, "A Normalized Encoder-Decoder Model for Abstractive Summarization Using Focal Loss," in *Proc. of CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 383–392, 2018. Article (CrossRef Link).
[78] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020. Article (CrossRef Link).
[79] D. P. Kingma and J. Lei Ba, "Adam: A Method for Stochastic Optimization," 2014. Article (CrossRef Link)
[80] Y. Kim, "Convolutional Neural Networks for Sentence Classification," *arXiv:1408.5882*, 2016. Article (CrossRef Link).

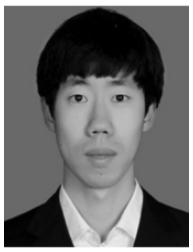


Wajid Ali received M.S. degree in Computer Science from the Department of Computer Science, University of Peshawar, Peshawar, Pakistan. He is currently pursuing Ph.D. degree with the College of Computer Science and Technology from Jilin University, Changchun, China. His current research interests include Natural Language Processing, Artificial Intelligence, Deep Learning, Causality Learning, and Recommendation.

Professor, Wanli Zuo received BS, MS, and Ph.D. in Computer Science degree from the College of Computer Science and Technology, Jilin University, Changchun, China. He is currently works as a Professor at the same institution. His main research interests include the Artificial Intelligence, Natural Language Processing, Deep learning, Classification, and Causality Learning.

Rahman Ali received Ph.D in Computer Engineering from Kyung Hee University, South Korea, an MS in Artificial Intelligence from University of Peshawar, Pakistan. Currently, he is working as an Assistant Professor at Quaid-e-Azam College of Commerce, University of Peshawar. His main research interests include Artificial Intelligence Data, Mining Machine Learning, Reasoning & Inference Recommendation System.

Gohar Rahman received his Master of Science in Computer Science (MSCS) from Agricultural University Peshawar, Pakistan in 2015. Currently, he is pursuing a Ph.D. degree in Information Technology (IT) with Universiti Tun Hussein Onn, Malaysia (UTHM). His research interest includes mutual authenticacaiton, Data mining, and data management.



Xianglin Zuo received the BS and MS degree from the College of Computer Science and Technology, Jilin University, Changchun, China. He is currently pursuing Ph.D. degree with the College of Computer Science and Technology from Jilin University, Changchun, China. His main research interests include the Artificial Intelligence, Representation Learning, Recommendation, Natural Language Processing, and Causality Learning.

Inam Ullah is currently an Assistant Professor in the School of Computer Science and Engineering of Shandong Jianzhu University. He obtained his Ph.D. in 2021 from the School of Software Engineering, Shandong University, Jinan, China. He received his bachelor's degree from the University of Peshawar, Pakistan, and his Master's degree from the International Islamic University of Islamabad, Pakistan. His current research interests include image processing, Computer vision, Machine Learning, deep learning, Salient Object Detection, and biometric recognition.