applied sciences

MDPI

Article

Toward a Multi-Column Knowledge-Oriented Neural Network for Web Corpus Causality Mining

Wajid Ali [1,2], Wanli Zuo [1,2,*], Ying Wang [1,2] and Rahman Ali [3]

1 College of Computer Science and Technology, Jilin University, Changchun 130012, China
2 Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China
3 Quaid-e-Azam College of Commerce, University of Peshawar, Peshawar 25000, Pakistan
* Correspondence: zuowl@jlu.edu.cn

Abstract: In the digital age, many sources of textual content are devoted to studying and expressing many sorts of relationships, including employer–employee, if–then, part–whole, product–producer, and cause–effect relations/causality. Mining cause–effect relations are a key topic in many NLP (natural language processing) applications, such as future event prediction, information retrieval, healthcare, scenario generation, decision making, commerce risk management, question answering, and adverse drug reaction. Many statistical and non-statistical methods have been developed in the past to address this topic. Most of them frequently used feature-driven supervised approaches and hand-crafted linguistic patterns. However, the implicit and ambiguous statement of causation prevented these methods from achieving great recall and precision. They cover a limited set of implicit causality and are difficult to extend. In this work, a novel MCKN (multi-column knowledge-oriented network) is introduced. This model includes various knowledge-oriented channels/columns (KCs), where each channel integrates prior human knowledge to capture language cues of causation. MCKN uses unique convolutional word filters (wf) generated automatically using WordNet and FrameNet. To reduce MCKN's dimensionality, we use filter selection and clustering approaches. Our model delivers superior performance on the Alternative Lexicalization (AltLexes) dataset, proving that MCKN is a simpler and distinctive approach for informal datasets.

Keywords: causality mining; knowledge-oriented channels; multi-level knowledge-oriented network; relation network; relation classification

check for updates

Citation: Ali, W.; Zuo, W.; Wang, Y.; Ali, R. Toward a Multi-Column Knowledge-Oriented Neural Network for Web Corpus Causality Mining. *Appl. Sci.* **2023**, *13*, 3047. https://doi.org/10.3390/app13053047

Academic Editor: Francisco García-Sánchez

Received: 13 January 2023
Revised: 19 February 2023
Accepted: 21 February 2023
Published: 27 February 2023

1. Introduction

Causality mining is an important method of artificial knowledge discovery that makes use of unstructured datasets. It now presents a crucial and unsolved challenge for NLP. Due to the underlying semantics, grammar, increasing vocabularies, and ambiguous nature of natural language text, causality mining remains a difficult job. As a result, it has prompted a great deal of academic interest in the last 10 years. The development of ML (machine learning) and DL (deep learning) methods has allowed academics to develop more productive models. Causality plays a significant role in decision making [1], question answering [2,3], relationship among everyday activities [4,5], event prediction [6,7], and generating future scenarios [8]. Causality exits in a wide range of disciplines including Environmental Sciences [9], Computer Science and Biology [10], Psychology [11,12], Linguistics [13,14], Medicine [15], and Philosophy [16]. Despite some similarities, the terms "causality" and "correlation" have different meanings. A correlation between two entities, however, does not always mean that a change in one thing is what caused the values of the other thing. The causality refers to the relationships between two regularly occurring events (e1 and e2) or phenomena (P1 and P2), i.e., that the existence of P1 or e1 causes the occurrence of P2 or e2. However, it is challenging to define the term causation in a broader

sense [17]. The idea has long been up for debate among philosophers. The sociological theory dictionary [18] provides the following standard definition:

- When one event or series of related events occurs first and paves the way for the occurrence of subsequent events. When the initial event (cause) occurs, the second event (effect) inherently or definitely follows.
- According to the theory of multiple causations, there are numerous possible causes for a particular event, each of which might be sufficient but not necessary for the effect to occur or necessary but insufficient for the effect to exist.

Causality mining emphasizes the automatic detection/extraction of causality between events in the text. As an example, "The 2019 COVID-19 pandemic caused a series of shocking deaths around the world" implies causality between the cause (COVID-19 pandemic) and the effect (deaths). Numerous methods for establishing causation are listed in the literature, including causal knowledge and causal discovery [19,20]. However, earlier attempts at causality mining relied on machine learning and rule-based methods. Rule-based methods, however, require carefully planned linguistic features [13,15,21,22]. Usually, these methods overlook hidden features and sequences of events. They are only able to mine domain-dependent explicit causality from phrases, and they do not take into account the features that point to the presence of dependence relationships. Similar to this, in machine learning approaches [4,23–25], semantic, syntactic, and lexical features are constructed by human operators through diligently feature engineering, and causality is automatically determined from large labeled datasets. The model's effectiveness in these techniques depends on the arrangement and regularities of its features. However, the absence of annotated datasets restricts these methods, which leads to error propagation in the systems.

Currently, the rising demand for the deep neural network such as RNN [26,27], CNN [28], MCNN [29], Transformer block [30], BERT [31], TinyBERT [32], and Hopfield neural network (HNN) [33–35] make it possible to perform various processing tasks without complex feature engineering. Deep networks play a key role in encoding the linguistic nature of words into fixed-size vectors to lessen the dependence on NLP toolkits [36] by using pre-trained word embedding. A key component of many NLP strategies, pre-trained word embedding offer a number of advantages over embedding taught from scratch [37]. However, due to the causality ambiguity and implicit nature in web corpora, it is still beyond the scope of DL techniques. To mine implicit and ambiguous causality in the web corpora, we proposed a novel deep MCKN model that outperforms the state-of-the-art in terms of its ability to manage the causality problem. However, most of the current approaches are strict when it comes to autonomously engineering features in implicit and ambiguous datasets. As a result, the proposed MCKN is based on a multi-knowledge-oriented channel by parsing every word in the source segments and connective (AltLex) and then identifying causation between the segments on both sides of the connective.

According to the proposed paradigm, each channel has a specific input segment or connective that presents a particular set of KCs. Each channel can incorporate linguistic knowledge of cause-and-effect relationships from world knowledge bases (WordNet, FrameNet) by capturing important linguistic cues of causality at the segment and connective level. Each channel uses a variety of convolutional word filters with different window sizes to create numerous unique feature maps.. Utilizing max pooling, the convolution results of each filter are further aggregated, and the feature map is mined for the most promising features that can be used for classification. The usage of "wf", a pre-trained word embedding generated automatically by Algorithm 1 from the "Bootstraps" corpus, lowers the overall dimensionality of the proposed model. After max pooling, the feature maps of each channel are combined, and dimensionality reduction is used to reduce the dimensionality of the combined feature maps. In the end, four object pairs were produced and sent to RN for further processing since RN needs object pairs for relation reasoning. For the same purpose, we also apply WordNet categorical approaches, "wf" selection techniques, FrameNet causal scores, and clustering algorithms for redundant and non-discriminative features. The goals and contributions of our research are described in the section below.

Algorithm 1: Automatic word filters' generation

***Step 1*:** Find all the lexical units of 50 causal semantic frames from FrameNet and group them by the number of words (max: 64).

$$lu_1 = \{C_1, C_2, \ldots, C_{n1}\}$$
$$lu_2 = \{[C_{11}, C_{12}], [C_{21}, C_{22}], \ldots, [C_{n_21}, C_{n_22}]\}$$
$$lu_3 = \left\{[C_{11}, C_{12}, C_{13}], [C_{21}, C_{22}, C_{23}], \ldots, \left[C_{n_31}, C_{n_32}, C_{n_33}\right]\right\}$$
$$lu_4 = \left\{[C_{11}, C_{12}, C_{13}, C_{14}], [C_{21}, C_{22}, C_{23}, C_{24}], \ldots, \left[C_{n_41}, C_{n_42}, C_{n_43}, C_{n_44}\right]\right\}$$
$$lu_5 = \left\{[C_{11}, C_{12}, C_{13}, C_{14}, C_{15}], [C_{21}, C_{22}, C_{23}, C_{24}, C_{25}], \ldots, \left[C_{n_51}, C_{n_52}, C_{n_53}, C_{n_54}, C_{n_55}\right]\right\}$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$lu_{64} = \left\{\begin{matrix}[C_{11}, C_{12}, C_{13}, C_{14}, C_{15}, C_{16}, \ldots, C_{164}], [C_{21}, C_{22}, C_{23}, C_{24}, C_{25}, C_{26}, \ldots, C_{264}], \ldots, \\ \left[C_{n_{64}1}, C_{n_{64}2}, C_{n_{64}3}, C_{n_{64}4}, C_{n_{64}5}, C_{n_{64}6}, \ldots, C_{n_{64}64}\right]\end{matrix}\right\}$$

Step 2 : *Extend lexical units of lu1 using WordNet*

for *word in* lu_1 ***do***
  for *synset in WordNet synsets of word* ***d***
    if $\left\{\begin{matrix}\text{"effect", "cause", "responsible", "causation", "result",} \\ \text{"reason", "because", "leadto", "dueto"}\end{matrix}\right\}$ *in WordNet gloss synet* ***then***
      for *lemma in WordNet lemmas of synet* ***do***
        if length of lemma == 1 **then**
          $lu_1 = lu_1 + lemma$
        else **if** length of lemma == 2 **then**
          $lu_2 = lu_2 + lemma$
        else **if** length of lemma == 3 **then**
          $lu_3 = lu_3 + lemma$
        else **if** length of lemma == 4 **then**
          $lu_4 = lu_4 + lemma$
        else **if** length of lemma == 5 **then**
          $lu_5 = lu_5 + lemma$
        ..... **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 KC convolutional weights.*

for *each lexical unit* $[c_1, \ldots, c_i]$ *in* $lu_j, (j = 1, 2, 3, 4, 5, 6, 7, 8, \ldots, 64)$ ***do***
  the corresponding filter weights is :
$$f = [f_1, \ldots, f_k]^T$$
  where $\boldsymbol{f_k} \in \boldsymbol{R^e}$ *is the word embedding of* $\boldsymbol{c_i}$ *found by looking up the word embedding table* $\boldsymbol{W^{word}} \in \boldsymbol{R^{e \times |v|}}$ *and* $\boldsymbol{k}$ *is the convolutional window size.*
end for

- This study is unique in that it analyzes sentences for causality leveraging web corpora, which include noisier, larger, and more muddled data.
- The suggested model uniqueness is its first-ever use of multiple KCs and a novel word filter technique, which significantly decreased the model dimensionality.
- The proposed model addresses implicit and ambiguous intra-sentence causality using segment and connective levels features.
- This is the first attempt to train in all channels by using convolutional "wf" rather than a data-oriented pre-defined convolutional filter.

- Extensive experiments on publicly available datasets have shown that the MCKN model performs much better than many baseline methods and text classification techniques.

The remainder of this article is structured as follows. Section 2 presents the literature review. Section 3 details the suggested strategy. The entire experimental process is covered in detail in Section 4. Finally, Section 5 summarizes our conclusion.

2. Literature Review

In terms of causality mining, previous research has mainly been divided into ML and DL methods. The performance gain of DL over ML techniques is significant. ML approaches normally require sophisticated feature engineering. For ML approaches, Ref. [38] uses a dependency structure to derive causation event pairs. In [39], causal connectives were used to govern how lexico-syntactic patterns and causal connectives interacted. These connectives were obtained by computing the similarity of sentence syntax-dependent structures through the Restricted Hidden Nave Bayes (RHNB) classifier. In [22], a related monolingual corpus of simple and English Wikipedia PDTB is utilized to integrate world knowledge (WordNet, VerbNet, and FrameNet) to evaluate the correlations across words and segments though hardly handling those terms that never occur in the learning stage. In [40], conditional text generation networks are proposed to craft possible causes and effects for any free-form textual event. They focus on explicit relations within individual sentences by linking one part of a sentence to another and using generated patterns instead of sentence-level human annotation.

In contrast to ML approaches, models in deep learning techniques automatically learn and extract useful features. In NLP, such models use pre-trained word embedding (Google News, GloVe-6B, GloVe-840B, and Pre-trained Wiki), which play a significant role in encoding syntactic and semantic properties of words into fixed-size vectors to reduce the dependency on NLP toolkits [36]. In NLP, two commonly used models are RNNs and CNNs. CNNs and RNNs [41] have been applied to document and paraphrase classification [42–44] and relation extraction/classification [36,45]. In [46], a variant of CNN, multi-column convolutional neural networks (MCNN) is presented to handle multiple features in the question answering (QA) of candidate answers. An analogue of MCNNs for relational classification is proposed in [47], as the piecewise max-pooling network. In [29], using external BK (background knowledge), a well-known model of MCNNs based on [48] is introduced. By utilizing question and response sequences [8], the MCNN model enriches causality attention, which is in contrast with [29].

In [49], the FFNN (feed-forward neural network) is proposed to augment the feature set to identify causality by computing the distance among events triggering words and related words in phrases. The work of [50] is closely related [49] by using a novel FFNN with a novel contextual word extension method. They use BK as an event context word extension to extract causal network structures from news articles to classify event causality. This is a challenging job as tweets often consist of a highly informal, unstructured nature, and lack contextual knowledge. In [51], a TCDF (temporal causal discovery framework) is presented to obtain a temporal causal graph by mining cause–effect relationship in time series datasets. They applied a multi-attention-based CNN with a causal support stage. BERT is a deep pre-trained language representation system using masked and Transformer blocks, which produced improved results in various NLP tasks, driven by transfer learning in computer vision.

In [26], a novel knowledge-oriented CNN (K-CNN) is presented for causal relations recognition, which combines a data-oriented channel (DOC) and a knowledge-oriented channel (KOC). The DOC acquires major features of causal relationships in the source data, while KOC adds human past knowledge to retain the linguistic clues of causal relationships. KOC automatically generates convolutional filters from FrameNet and WordNet without the requirement to train a classifier with a lot of data. Such filters are causal word embedding. In contrast to statistical, non-statistical, and single-level DL models, deep multi-level models exhibit satisfactory performance. However, they hardly incorporate implicit and

ambiguous causality. In [52], a graph reasoning technique based on document-level context is proposed to recognize event causality. In [53], a SCITE (self-attentive Bi-LSTM-CRF wIth Transferred Embedding) method is presented and formulates causality as a sequence tagging by mining causal event pairs and their relationship. Moreover, they use multi-head self-attention to enhance their performance [30].

In [54], a novel approach is proposed that exploits the advantages of neural model-based approaches and feature engineering. The latest work [55] uses a head-to-tail entity annotation method that expresses the entire semantics of complex cause–effect relationships and visibly finds entity boundaries in source sentences. They employ entity location perception along with RPA-GCNs (Relation Position and Attention-graph Convolutional Networks), GATs (Graph Attention Networks), and other techniques. In [56], a generative approach for extracting cause–effect relationships via encoder–decoder and pointer networks. They enhanced the performance but required more time to produce the required result. In comparison to statistical and non-statistical techniques, DL approaches with pre-trained word embedding are more fruitful. However, they work on a huge training dataset that covers all causality expressions in the source text, which is somewhat impossible due to the diversity and ambiguity of phrases and words in the dataset. However, the ambiguous and implicit nature of causality is a challenging task. Our MCKN is motivated and inspired by [26,29,31,57] for mining implicit and ambiguous causality sentences from publically available web corpora. To begin, keep in mind that previous methods for leveraging MCNNs for NLP applications used multiple CNN channels with pre-defined convolutional filters for training. Our inspiration is a novel approach using the concept of the MCNN approach. Contrary to MCNNs, the proposed approach is based on knowledge-oriented channels by using novel convolutional word filters generated by Algorithm 1. Table 1 provides a more concise description of the reviewed material.

Table 1. A list of related techniques that have been examined.

References	Description	Features	Benefits	Drawbacks
[38]	Using a supervised approach for extracting causality	Using event 1 cause, event 2 patterns	First attempt toward web corpus	Focused on explicit domain- specific causality
[22]	Mining AltLexes of causal discourse relations	Using Explicit discourse connectives	Target web coups and creation of AltLexes dataset	Working with a large feature vector set typically slows down the processing of the model
[29]	An approach based on MCNNs [48]	Using prior BK features	Targeted implicit and ambiguous causality	Japanese dataset was emphasized
[39]	RHNB classifier	RHNB model based patterns	Combine lexi-cosyntactic pattern and causality connective in one place	Target large features that typically slow down model performance
[40]	Conditional text generation network	Using generated patterns	Bypass human effort	Emphasized explicit relationships within individual sentences
[8]	Attention based MCNNs	MCNNs + Attention	Inspiration toward causality attention	For QA using causality attention
[49]	Computing the distance among events triggering words and related words in phrases events	FFNN based	They targeted implicit and ambiguous causalities	Over-fitting problem

Table 1. *Cont.*

References	Description	Features	Benefits	Drawbacks
[50]	A novel model with an event context word extension mechanism	FFNN + BK	Targeted implicit causalities social media tweets	Results in info loss due to opinionated posts
[51]	TCDF model	Considered multi-attention-based CNN with a causal support stage	Centred on implicit observational time series data	Executes rather worse on short time series
[26]	A novel K-CNN for causality extraction	Combined DOC and KOC.	Focused on implicit causality	Over-fitting problem
[53]	SCITE with multi-head self-attention [30]	They formulate causality as a sequence tagging	✓	✓
[54]	A deep neural based MCDN approach	Using word and segment level information	Focused on implicit and ambiguous	They only looked at causality within sentences and ignored multi-level sentences
[56]	Generative approach	Using encoder–decoder and pointer networks to extract causality	Focused on implicit causality	Limited to financial dataset

3. Proposed Approach

This section explores the MCKN model. This model consists of three channels/columns, where each channel deals with its respective AltLex/connective (L), segments after AltLex (AL), and segments before AltLex (BL) in the target sentence. MCKN mainly targeted implicit and ambiguous causalities. The MCKN uses convolutional word filters instead of pre-trained convolutional filters. In Figure 1, we explored the architecture of the MCKN model, including (i) the first column dealing with the BL (e1) part of the input sentence, (ii) the second column dealing with the L part of the input sentence, and (iii) the third column dealing with the AL (e2) part of the input sentence. More details about Figure 1 are covered in Section 3.4.

3.1. Linguistics Background of Source Corpus

This part discusses the linguistic background of causality and the AltLexes (https://github.com/chridey/altlex, accessed on 3 May 2021) dataset. About 12% of the Pine Discourse Tree Bank (PDTB) is labeled as causal, and around 26% is implicit [58]. In addition, there exists another type of implicit relation called "AltLex", which represents causality and is marked as an open and infinite class of causality. The generalization of "AltLex" is extended with an open class of markers [22]. Some examples in the "AltLexes" dataset are not present in the explicit relations of PDTB including ambiguous causal verbs, e.g., "COVID-19 made many countries affected" and partial prepositional phrases, e.g., "He has made aircraft with the idea of a new deep neural technology". In the first example, the term "made" has numerous meanings and is employed to express causation. However, in the second example, the causal relationship expression is not clear. According to our analysis, the parallel data constructed has 1164 causal connectives and about 7627 non-causal connections. Furthermore, their intersection has 155 types of connectives, which are hybrid. It shows their reliance on a causal set of 12.6%, and reliance on non-causal sets is 1.8% [22]. Several implicit and heterogeneous relationships are discovered as a result of the analysis. In this case, prior approaches have several demerits to making an expert system.

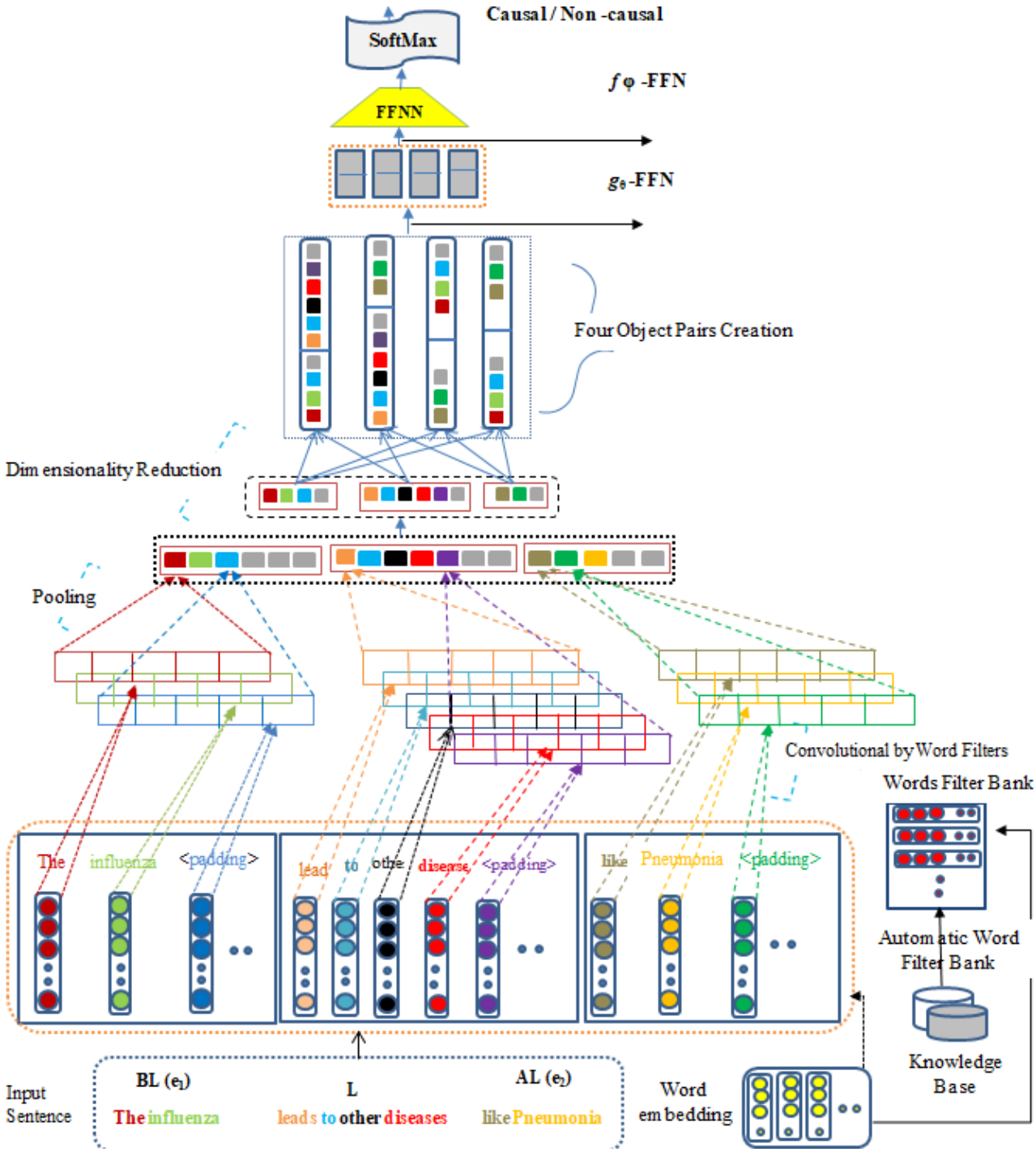

Figure 1. Applied MCKN architecture. This contains three columns using convolutional word filters: the first column processes the BL (e1) part of the input sentence, the second column deals with the L part of the input sentence, and the third column targets the AL (e2) part of the input sentence.

3.2. Input Sentence Representations

The input sentence (N) contains 'n' tokens, $N = \{n_1, n_2, \ldots, n_{i-1}, n_i\}$. Where '$n_i$' is the filter token in the sentence at 'i' position. Further, each sentence is formatted to L, AL, and BL. The purpose is to generate sentence level 'y' predication, where 'y' is the input sentence label shown in Equation (1). For the parallel corpus feature in our model, we employ a pair

of simple and English Wikipedia sentences, although it still only takes a single sentence as input [43].

$$y = \begin{cases} 1, & |Causal\ sentence \\ 0, & |Not - Causal\ sentence \end{cases} \tag{1}$$

Motivated by [30], each token/word in the input sentence can be denoted by summing the corresponding token embedding, position embedding, and segment embedding. Likewise, the early work segments embedded here indicate segments L, BL, and AL in each sentence. In Figure 2, first of all, the "word2vec toolkit" is used for pre-training of word embedding with dimension d_{word}, positional embedding with dimension d_{pos}, and segment embedding with dimension d_{seg} for linguistic information. Lastly, summing all three embeddings results in new representation $\acute{N} = \{z_1, z_2, z_3 \ldots, z_{n-1}, z_n\}$, where $z_n \in R_d$ for token n_i, and keep equal $d = d_{word} = d_{pos} = d_{seg}$ dimensions of the word embedding, position embedding, and segment embedding. Therefore, the $\acute{N}$ representation of input sentences could bring fundamental features to complicated networks.

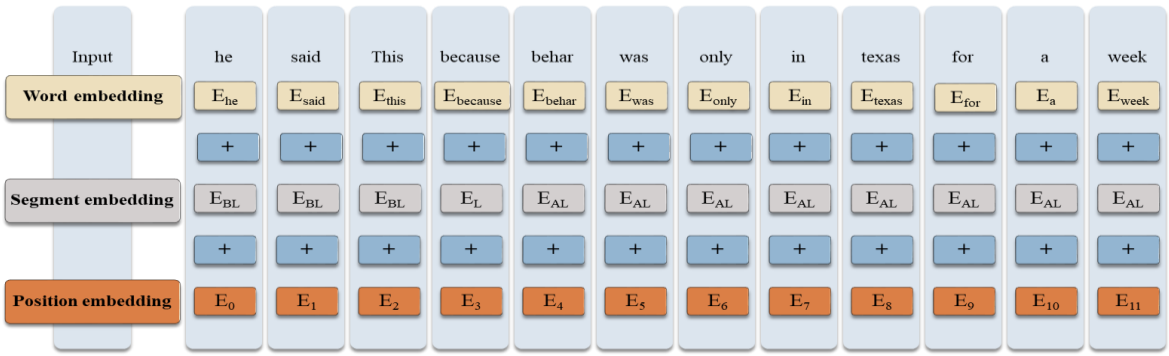

Figure 2. Input representation for MCKN. The word, segment, and position embedding are added to create the input embedding.

3.3. Relation Network

In visual question answering (V-QA) [59], the relation network (RN) plays a very significant role. It can be efficiently integrated with DL approaches including CNNs (DeepCNN, knowledge CNN, and MCNN) and RNN (GRU, bi-GRU, LSTM, bi-LSTM) for performance enhancement. The original RN, however, only performs single-step inference, such as $A \rightarrow B$ rather than $A \rightarrow B \rightarrow C$. For those tasks which need multistep relational reasoning, Ref. [60] introduced RNNs that work on graph representations of entities.. Furthermore, Ref. [61] made memory networks with RNs capable of complicated reasoning, which changed the computational complexity from nonlinear to linear. Though, most jobs are only used for text and V-QA. Similarly, we consider RN in the proposed model, which takes input object pairs from KCs and makes effective relational reasoning.

3.4. About Knowledge-Oriented Channel

Encouraged by [26], we have applied three different KCs to recognize keywords, cue words, and cue phrases of causality in connective and segment levels of the input sentence. For convolution operation, we used "wf" in each channel, which is a variant of the convolutional filter. It is automatically generated from knowledge bases (WordNet, FrameNet) using the linguistic knowledge of causality. Compared with the CNN convolutional filters, the "wf" more precisely represents causal relationships. The weights of "wf" are the pre-trained word embedding, which can be used without additional training. Using the "wf" approach will significantly drop the number of pre-parameters of the model and reduce the over-fitting issue in the small data corpus. Figure 1 depicts the proposed network's architecture, which consists of three channels. Each channel has its specific input segment format including segments BL, AL, and connective L. The "L" is usually used to represent the cue phrases, cue words, and keywords for cause–effect relations.

Examples of such connectives include because, as result, lead to, resulted, due to, and trigger. These words in the connectives part away from the BL and AL segments and may affect the performance of a network. In the past, KCs only paid attention to the "L" part of the input sentence (between event e1 and event e2) because it usually represents causality signals. However, in the proposed model, each segment and connective has its own KCs. To decrease the morphological variations of tokens in each segment, we used WordNet tokens to make it consistent and mark each word in its lowercase by using the lemmatizer function, and further, every word is converted into a precise input format as shown in Figure 2. The single knowledge-oriented channel of the proposed model is shown in Figure 3. In the input format, we set the maximum size of L as 8 words, and each BL and AL to be 64 words. Sentences with fewer than 8 words in the "L" level and fewer than 64 words in the segments level are padded with padding characters with zero embedding because CNN works with a fixed input size.

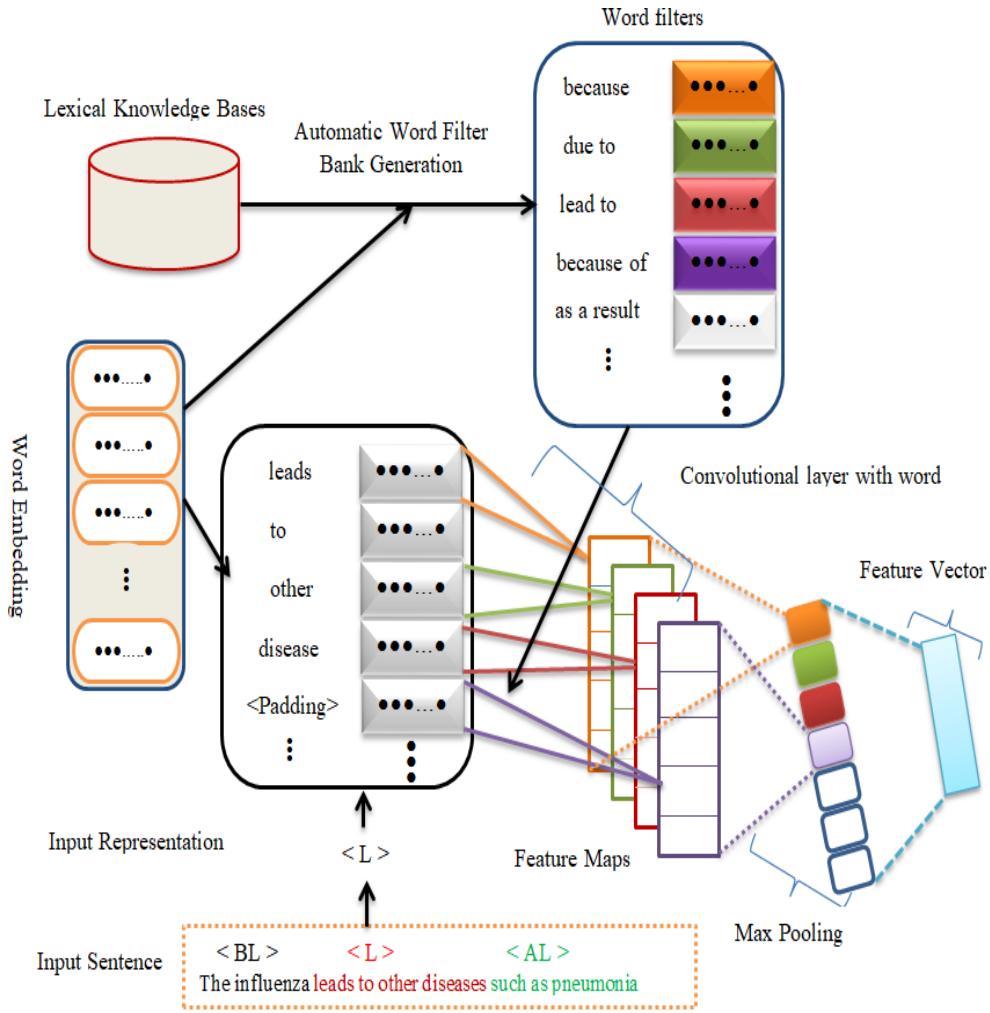

Figure 3. Knowledge-oriented channel for connective part of the input sentence.

Word Filters Archive Generation

Word filters are the embedding of causal words, cue words, and cue phrases that are extracted from WordNet and FrameNet knowledge bases. Among them, WordNet is a huge database of lexical that categorizes English words into sets of synonyms known as synsets to denote diverse concepts [62]. To mimic their semantic and lexical relationships, all synsets are linked in a hierarchical format. The meaning of every synset is given by a gloss with some examples [26], where example 1 describes the WordNet elements for the word "cause", which belongs to specific synsets. Similar to this, FrameNet is a lexical

resource built on the frame semantic theory; it organizes English phrases and words into higher-level semantic frames exploring a variety of ideas [63]. Each frame is a conceptual arrangement that includes a discussion of the type of event, the relation, or the object with a conceptual definition; the participants in the frame are called frame elements, words that frequently appear in the frame (referred to as lexical units (Lu)), and the relationship to other frames. The FrameNet components of the "Causation" frame are described in Example 2 [26].

In the proposed work, 50 causal frames (CF) are identified from FrameNet including triggering, response, causation, causation_scenario, reason, and explaining the_facts, and also 44 frames starting with the word cause. The "Lu" involved in this CF is the important clues and regularly seemed words that raise causality in the text, hence these "Lu" can be preserved much like cue phrases, clue words, and keywords of causality. To further extend these "Lu" to cover causal words more widely, we automatically construct a bank of causal words. These causal words and word embedding are used to find the weights of convolutional "wf". Automatically generating "wf" is accomplished by utilizing the improved Algorithm 1 [26]. Such "wf" more efficiently represents keywords, clue words, and cue phrases of causality. These "wf" are more effective than the convolutional filter learned from training. Moreover, the weights of these "wf" are static values.

Finally, about 850 uni-grams, 240 bi-gram, and 20 tri-gram "wf" are created. During convolutional, "wf" is convolved with n-grams to obtain the important linguistic clues of causal relationships in the input text, resulting in a sequence of similarity scores. The proposed convolutional method is capable to capture semantically related causal words other than those that exist in the "wf" bank. We create several different filters for the L part of the input sentence, where each filter size ranges from 1 to 8 filter words. Similarly, we create different filters for AL and BL; each filter size ranges from 1 to 64 filter words. The convolutional "wf" for every "Lu" is formatted as $[c_1, c_2, \ldots, c_i]$ in Lu_j, $(j = 1, 2, 3, 4, 5, 6, 7, 8 \ldots 64)$, the weights of corresponding 'wf' are $f = [f_1, f_2, \ldots, f_k]^T$. Where $f_k \in R^e$ is the word embedding of c_i discovered by looking the word embedding table $W^{wrd} \in R^{e \times |v|}$. Further, the $f = [f_1, f_2, \ldots, f_k]^T$ convolved with input text matrix $emb_k = \{w_1, w_2, \ldots, w_{n_1}\}$, where k is the convolutional window sizes (uni-gram, bi-gram, and n-gram). We follow [26], and modify the convolutional operation of each KCs so that each 'wf' becomes a future map $m = \left[m_1, m_2, \ldots, m_{n_1,-k+1}\right]$, where m_i signifies the similarity among the "wf" and the k-gram $w_{kgram} = [w_1, \ldots, w_{i+k-1}]^T$ in input sentence. The improved convolutional method is represented by Equation (2).

$$m_i = \left(\sum_{j}^{k} f_j^T w_{i+j-1} + b \right) / k \tag{2}$$

In Equation (2), "b" represents the bias term. Rather than using a non-linear function, we divided the CNN convolutional results by the window size k. By limiting f_j and w_{i+j-1} (word embedding) to unit vectors, the resultant value of m_i becomes the cosine similarity between f and w_{kgram}. The goal of cosine similarity in feature maps is to achieve equal importance of "wf" with different lengths by creating the same scale for all convolutional window sizes, while the conventional method will obtain a higher number for the wider window size. The most specific feature map is generated using max-pooling for each filter to further aggregate the convolutional results. The pooling procedure for each feature map is shown in Equation (3). The largest cosine similarity provides strong cues for the presence of cue phrases and keywords in the text, which is why the feature map maximum value is obtained.

$$p = max\{m_1, m_2, m_3, \ldots, m_{n_1-k+1}\} \tag{3}$$

About 900 "wf" are produced by Algorithm 1, which is thought to be highly dimensional and has limited training data. These "wf" for causality mining provide a large number of features, some of which may be redundant and irrelevant. In order to enhance the performance of the model, we used "wf" clustering and selection [26].

3.5. Segments and Connective Level Processing

The proposed model presents a novel method to mine causal relationships within a single sentence at the connective and segment level using 3 KCs and RN. The input connective L and segments BL and AL can be denoted as $Z_L \in R_{J_{L\times d}}$, $Z_{BL} \in R_{J_{BL\times d}}$, and $Z_{AL} \in R_{J_{AL\times d}}$ input format. Where, J_{BL}, J_{AL}, and J_L are the token lengths in each segment and connective. Each channel is responsible for parsing Z_{BL}, Z_L, and Z_{AL} into a set of objects. Unlike [26,58], MCKN convolves them through a 1D convolutional layer into different window sizes for "k" feature maps of size $J_{BL\times 1}$, $J_{L\times 1}$, and $J_{AL\times 1}$, where "k" is the sum of "wf". After convolution, each segment's and connective feature maps are rescaled into a k-dimensional vector via a max-pooling layer, and dimensionality reduction is then implemented by further reducing the dimensionality. Finally, we create a set of objects in Equation (4).

$$\left\{o^{BL}, o^{L}, o^{AL}\right\} \in R^k \tag{4}$$

In addition, because RN works with objects, we created four object pairs in Equation (5).

$$ObjectPair = \begin{vmatrix} o^{BL}; o^{L} \\ o^{L}; o^{AL} \\ o^{BL}; o^{AL} \\ o^{AL}; o^{BL} \end{vmatrix} \tag{5}$$

The ";" is now an operator that concatenates object feature vectors. We can simplify it using the notation in Equation (6), where '$*$' represents a pair-wise operation. For causality candidates, $BL * L$ and $L * AL$ determine the relationship between the cause-effect event and L, while $BL * AL$ and $AL * BL$ infer the direction of causality.

$$o^{BL} * L = \left[o^{BL}; o^{L}\right] o^{L} * AL = \left[o^{L}; o^{AL}\right] o^{BL} * AL = \left[o^{BL}; o^{AL}\right] o^{AL} * BL = \left[o^{AL}; o^{BL}\right] \tag{6}$$

As a result, the simplified form of the object is represented by the Equation (7).

$$Op = \begin{vmatrix} o^{BL} * \mathrm{L} \\ o^{L} * \mathrm{AL} \\ o^{BL} * \mathrm{AL} \\ o^{AL} * \mathrm{BL} \end{vmatrix} \tag{7}$$

Here $Op \in R^4 \times (2k + 2dg)$ is the matrix representation of object pairs. More generally speaking, by changing the architecture in a mathematical formulation, we were able to derive the final representation (Final_rep ϵ R^{4dg}) at the segment and connective levels in Equation (8).

$$Final\_rep = f_{\Phi}\left(\sum g_{\theta}(Op)\right) \tag{8}$$

At the segments and connective level, MCKN transforms segments and connectives into object pairs and then integrates these object pairs for pair-wise inference to discover the relationship between segments and connectives.

3.6. Causality Identification

The applied model discovers causality in each sentence by passing "Final_rep" to FFN. We used a 2-layer FFN involving a "dg" unit with a ReLU function followed by SoftMax for prediction, which is expressed mathematically in Equation (9).

$$\mathrm{FFN}(\mathrm{Final\_rep}) = \mathrm{SoftMax}(\mathrm{ReLU}(\mathrm{Final\_rep}\,(W_1 + b_1)W_2 + b_2) \tag{9}$$

There is rich discrimination between causal and non-causal samples in the AltLexes dataset. By using a Cross-Entropy (CE) loss function, the apparent inequality of causal and non-causal examples in the source dataset can lead to unsatisfactory outcomes. Since each connective and segments in the target sentence contain an ambiguous and heterogeneous connective (make, made, create, construct, etc.), effect keyword (disable, lost, miss, destruc-

tion, company, died, etc.), and causal keywords (lack, accident, fire, tupan, tsunami, flood, earth quick, blast, etc.), it is hard to detect in each sentence.

$$Lfl = \begin{cases} -a(1-\hat{y})\beta & y = 1 \\ -a(1-\hat{y})\beta & y = 0 \end{cases} \tag{10}$$

As a result, it is required to give causal and non-causal losses a soft weight, enabling the model to focus more on ambiguous, implicit, and heterogeneous samples. Inspired by [54,64], we consider the focal loss into a progress loss function [65], by adding a modulating factor to the CE loss $(1-\hat{y})\beta$, with a tunable hyperparameter $\beta \geq 0$. In Equation (10), the focal loss L_n is formulated as the objective function, with 'α' denoting the balance weight hyperparameter.

4. Experimental Settings

In this part, we explore the MCKN model at the sentence level, which combines three knowledge-oriented channels for causality mining.

4.1. Datasets

We used the "AltLexes" web corpus [22] in our proposed technique, which consisted of 86,896 training samples, of which about 7606 are causal, and 79,290 are non-causal samples, a bootstrapped set of 100,744 samples, of which about 12,534 are causal and 88,240 are non-causal. The bootstrapped dataset is produced using new AltLexes to bootstrap to find more examples, which increases the causal samples by about 70 percent. The Dev set of 488 samples, of which about 181 are causal and 307 are non-causal, and the test set contains 611 samples, of which about 315 are causal and 296 are non-causal. We train MCKN on the bootstrapped and training sets separately and fine-tune them on the Dev set. Finally, the model is tested using the test set.

4.2. Hyperparameters and Evaluation Metrices

Hyperparameter: In the implementations, we set the initial learning rate of the proposed model as $1 \times 10^{-2.5}$, and gradually compress after the F1 score has stopped growing for more than 6 epochs. During training, we set the batch size to 32, the epoch size is 15, and apply L2 regularization to deal with the over-fitting issues with a 0.5 dropout rate. We set the regularization coefficient to 3×10^{-5}. For focal loss, we used $\alpha = 0.80$ and $\beta = 4.5$. For optimization purposes, Adam optimizer [66] is used with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\in = 1 \times 10^{-8}$ hyperparamenter and clipped gradients norm. We used $k = 130$ for the number of kernel/wf of various window sizes ranging from 1 to 8 at the "L" level and 1 to 64 at each of the "AL" and "BL" levels. In Table 2, we summarize all hyperparameters, which provides a more convenient approach for the reader.

Table 2. Hyperparameters and their values used by proposed work.

S.No.	Hyper Parameters	Values
1.	Learning Rate	$1 \times 10^{-2.5}$
2.	Batch Size	32
3.	Epoch Size	15
4.	Regulation function	L_2
5.	Dropout Rate	0.5
6.	Regularization Coefficient	3×10^{-5}
7.	Focal loss	$\alpha = 0.80$, $\beta = 4.5$
8.	Adam Optimizer	$\beta_1 = 0.9$, $\beta_1 = 0.999$, $\epsilon = 1 \times 10^{-8}$
9.	Kernel/Word filter Size	K = 130

Evaluation Metrics: We compare MCKN with the baseline techniques in Table 3 using a variety of evaluation metrics, such as precision, recall, and F1-score. The prediction ability of algorithms is measured by their precision (*Pr*). It illustrates how many positive predictions are achieved and how accurate predictions are made by individuals who make them. The '*Pr*' is calculated in Equation (11). Among these, true positive (*TP*) is the number of correctly classified positive cases, and true negative (TN) is the proportion of correctly classified negative events. False positive (*FP*) refers to the number of positively classed instances that were misclassified, while false negative (*FN*) refers to the number of positively classified instances that were incorrectly classified.

$$Pr = \frac{TP}{TP + FP} \tag{11}$$

Table 3. Performance comparison of MCKN with state-of-the-art approaches.

Models	Training Dataset			Bootstrapped Dataset		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
K-CNN [26]	80.57	43.17	53.54	78.45	73.97	76.14
DPCNN [57]	69.06	58.10	63.10	79.66	74.60	77.05
BERT-base [31]	68.97	63.49	66.12	75.99	85.40	80.42
MCNN [29]	85.13	**83.70**	83.90	80.46	90.15	87.51
MCKN	**90.55**	83.22	**86.81**	**93.13**	**91.79**	**90.15**

F-score (F1) is a crucial need for simulating the situation with the highest probability of obtaining the correct answer and explicitly demonstrating the algorithm's ability. Moreover, F1-score is defined as a harmonic mean of sensitivity and precision. The *F* value is calculated in Equation (12).

$$F = \frac{2TP}{2TP + FP + FN} \tag{12}$$

Recall (*Rc*) or sensitivity examines how well a case accurately yields a positive outcome for an instance that has an explicit condition. Equation (13) calculates the value of *Rc*.

$$Rc = \frac{TP}{TP + FP} \tag{13}$$

4.3. Baseline Methods

In this section, various baseline approaches are listed, including MCNN, K-CNN, DPCNN, and BERT-base. DPCNN [57] is a word-level deep neural network for topic categorization and sentiment classification. It can create downsampling without increasing the number of feature maps, which can efficiently represent long-range relationships. A deep pre-trained language representation system called BERT-base [31] is built on masked and Transformer blocks, and it has improved a number of NLP applications, encouraged by transfer learning from the computer vision sector. The next notable work in this field is MCNN [29], a multi-column CNN with BK that integrates event causality candidates and their contexts with relative web corpus. K-CNN [26] is the next novel work, which combines a data-oriented network with a knowledge-oriented network by using convolutional "wf", thereby reducing the overall dimension of the model.

4.4. Results

Before releasing the results, we run each reproducible experiment six times for causality extraction using a train/bootstrapped/Dev test split described in Section 4.1. Then, we report the average result along with its standard deviation. Table 3 compares MCKN's performance with state-of-the-art methods employing precision, recall, and F1-score in the test set, which is a randomly selected subset of both the train and bootstrapped datasets.

Our model performs, in particular, by learning distinct semantic representations of causation at the connective and segmental levels. In the train dataset, compared with the best state-of-the-art feature engineering methods [26,29,31,57], MCKN enhanced the maximum precision by 21.58% and a minimum of 5.42%, F1-score recorded by a maximum of 33.27% and a minimum of 2.91%, and similarly, a low recall rate is recorded 0.48%, since it emphasizes on the interchangeability of connectives, whereas parallel examples frequently contain the same connectives that might be evaluated as false negatives.

It is amazing that the proposed work on the bootstrapped train dataset enhances the precision up to a maximum of 17.14% and a minimum of 12.67%, the F1 score of a maximum of 14.01% and a minimum of 2.64%. They recorded a maximum of 17.82% recall and a minimum of 1.64% because the bootstrapped train dataset has many more samples of the causal signal compared to the training train dataset. The suggested model uses a novel combination of KCs with "wf," RN, and FFNN, as well as a unique combination of a different hyperparameter employed in the training stage, to achieve the best precision, recall, and F1-Score.

Contrary to CNN techniques such as K-CNN [26], DPCNN [57], BERT-base [31], and MCNN [29] with a pre-trained convolutional filter mechanism, the usefulness of the MCKN model is the uses of novel KCs with "wf". To the best of my knowledge, this is the first attempt to mine implicit causality in the web corpus using all KC channels with the unique "wf." Since "wf" may effectively target the causal relationships in the target sentence by effectively decreasing the number of parameters of the model. The proposed model performs satisfactorily when applied to single-sentence texts, but it is challenging to apply to texts with multiple sentences. The suggested model's successful findings demonstrate that deep knowledge-oriented convolutional techniques are more effective than conventional rule-based, statistical, and convolutional techniques in this area. Contrary to text classification, classifying causality is a challenging task that necessitates strong multi-level relational reasoning abilities. Figure 4 shows the relationship between epochs and their performance on the train dataset, while Figure 5 demonstrates the relationship between model performance and the number of epochs in the bootstrapped train dataset.

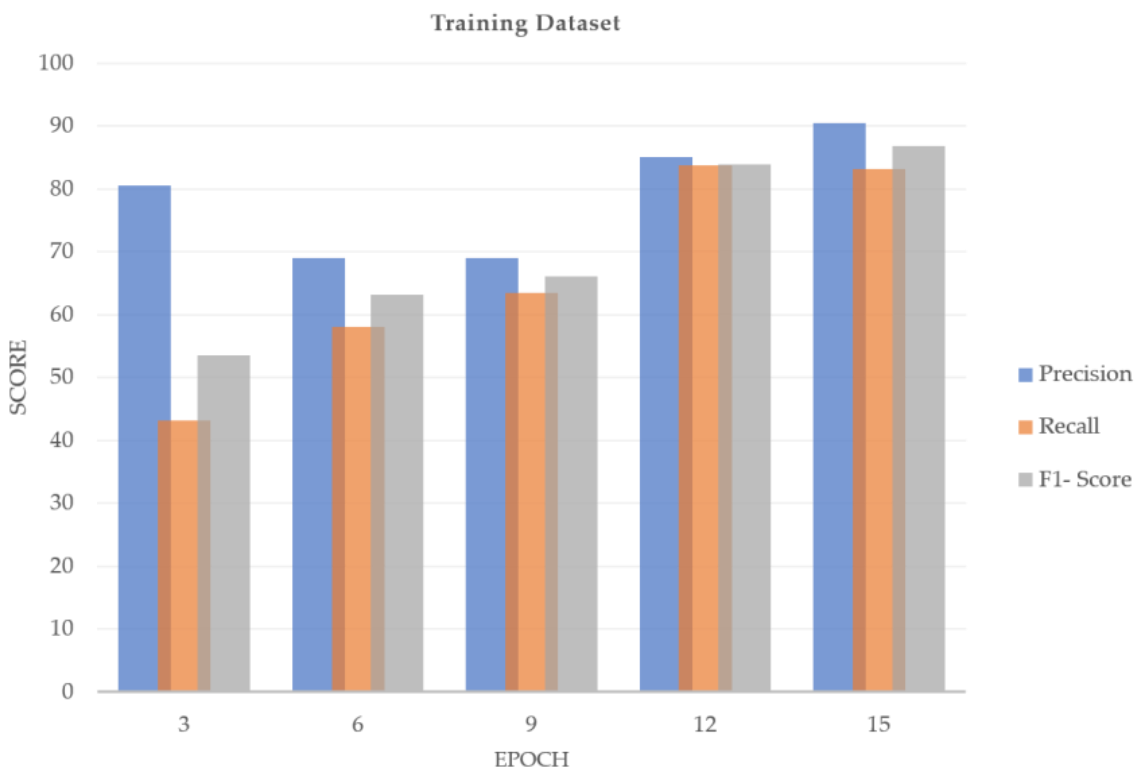

Figure 4. Train dataset: relationship between number of epoch and performance.

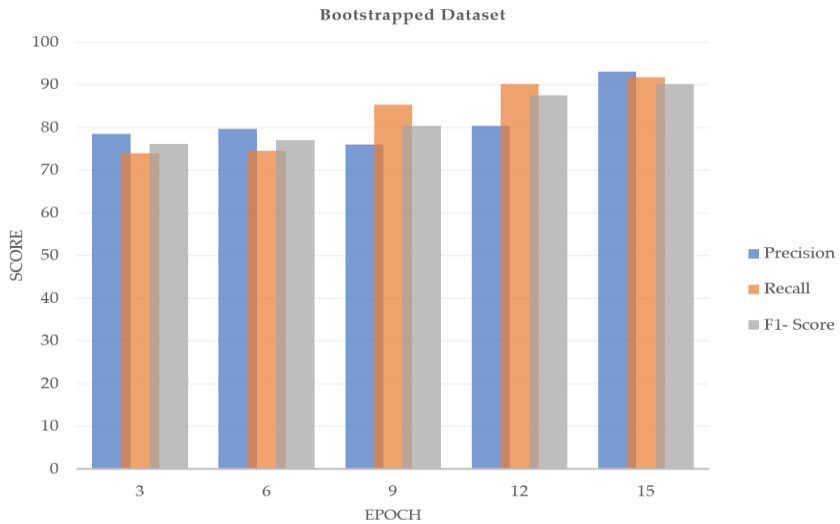

Figure 5. Bootstrapped dataset: relationship between number of epoch and performance.

4.5. Analysis

4.5.1. Effect of Multi-Column KNN

The validation matrices for the "AltLexes" dataset are shown in Table 3. We discovered from the conventional KNN models that the two-column KNN performs slightly better than the single-column KNN because it makes use of multiple convolutional window sizes, which can capture more information on causality from various n-grams. Similarly, by adding more information, the three-column KNN is better than two-column KNN. Contrary to K-CNN [26], we present three KCs together with RN. The development of the experimental results proves that multi-column KNN with convolutional "wf" can more effectively extract causality. The performance advantage of multi-column KNN over multi-column conventional CNN can be attributed to the following evidence:

- Compared with randomly initialized convolutional filters, the "wf" has an extra precise illustration and pays more extensive attention to the cue phrases, cue terms, and keywords of causality; this makes it possible for the model to more effectively extract linguistic cues that indicate causation in a sentence.
- The use of multi-channel KNN keeps the model from losing key causality properties. By utilizing already existing knowledge bases, the KCs are able to identify substantial language cues of causation at the connectives and segment level of the target sentences.
- In contrast to convolutional CNNs, KCs have a significantly lower pre-parameter count. This assists in resolving the issue of excessive over-fitting in a limited training dataset.

4.5.2. Strength of the MCKN

To understand more information about MCKN, we used both Areas under the Precision–Recall Curve (AUPRC) and Areas under the Receiver Operator Curve (AUROC) to estimate the specificity and sensitivity of the model. We evaluated the impact and robustness of different word embedding on performance. In the past, most tasks were based on a one-hot encoding and word-piece algorithm, different from pre-trained word embedding (GloVe-840B, Pre-trained Wiki, and Google News), used by our model. Table 4 shows pre-trained word embeddings with AUPRC and AUPOC scores, demonstrating the effectiveness of the proposed model.

Our drawing in Figure 6 more effectively illustrates the analysis of the pre-training words. In Figure 6, y-axis signifies the score of the Precision-Recall Curve (AUPRC), Areas under the Receiver Operator Curve (AUROC), and F-Score to estimate the specificity and sensitivity of the model.

Table 4. Performance analysis of pre-trained-word embedding.

Types	AUROC	AUPRC	F1-Score
GloVe-840B	87.48	88.58	83.50
Pre-trained Wiki	85.32	84.11	**85.23**
Google News	86.30	87.37	83.90
Average	86.80	87.29	84.90

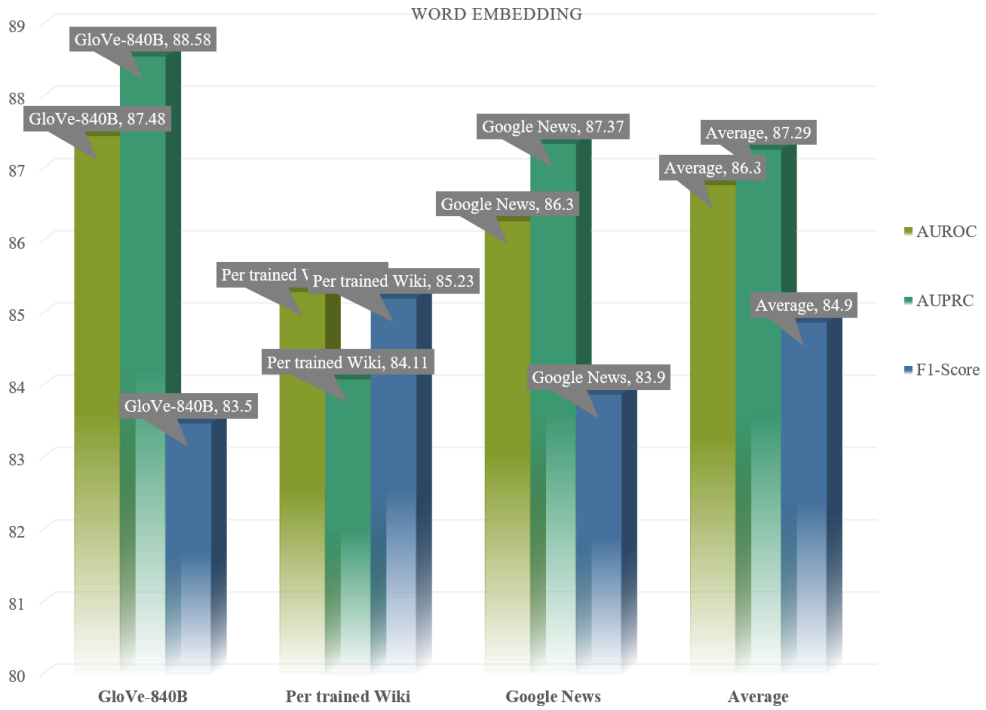

Figure 6. Analysis of different word embedding.

4.5.3. Ablation Study

Exploring MCKN and its contributions is very important to readers. In this section, we show the ablation evaluation through different training modules of the proposed model. Table 5 describes the results of the different modules of MCKN on the two datasets. In the training dataset, the single-column KNN + RN module reaches the precision (p) value of 78.74, the recall (R) value is 76.56, and an F1 Score (F-1) is 73.85. The two-column KNN+RN has enhanced the p value by 4.38, the R-value by 2.77, and the F-1 value by 7.54; further, the three-column KNN + RN module enhanced the p value of 7.43, the R-value of 3.89, and the F-1 value of 5.42 compared to two-column KNN + RN. Similarly, in the bootstrapped dataset, the single-column KNN + RN module reached a p value of 79.23, an R-value of 82.11, and an F-1 of 80.21. In the two-column KNN + RN, the p value is enhanced by 7.96, the R-value by 2.11, and the F-1 value by 5.6, of which the three-column KNN + RN module further enhanced the p value of 5.94, the R-value 7.57, and the F-1 value 4.34 compared to two-column KNN + RN. Based on the above analysis, compared with the single KNN + RN and two-column KNN + RN, the three-column KNN + RN (MCKN) shows significant results, because the three-column KNN + RN uses the combined features and knowledge of all channels. This demonstrates how multi-column KNNs and RN significantly boosted the model's overall performance.

Table 5. Ablation study of the proposed work.

Datasets	Metrics	Single-Column KNN + RN	Two-Column KNN + RN	Three-Column KNN + RN
Training Dataset	P	78.74	83.12	90.55
	R	76.56	79.33	83.22
	F-1	73.85	81.39	86.81
Bootstrapped Dataset	P	79.23	87.19	93.13
	R	82.11	84.22	91.79
	F-1	80.21	85.81	90.15

5. Conclusions

The novelty of this work is how to recognize ambiguous and implicit causality in the informal "AltLexes" web corpus. When compared to online corpora, the majority of earlier works used more formal newspaper, historical stories, and book corpora that incorporated clear causation. They frequently employ feature-driven supervised techniques to target explicit causality and overlook the implicit and ambiguous causation in the web corpus. In this work, a novel MCKN model is proposed that combines more than one KC and is integrated with RN for causality extraction in the unstructured web corpus. MCKN deals with each sentence at the connective and segment level for causal relational reasoning. The proposal employs a new convolutional word filter approach that drastically reduces the number of model parameters. Our model demonstrates the power of inferring complicated causation at the sentence level, in contrast to causality and document classification algorithms. Although, implicit and ambiguous causality and their event pair detection across sentences/multi-sentence text is still a demanding problem. For such task in future development, it is imperative to employ this model with more advanced features and a standardized dataset.

Author Contributions: Conceptualization, W.A.; Methodology, W.A.; Software, W.A.; Formal analysis, W.A., W.Z., Y.W. and R.A.; Investigation, W.Z.; Writing—original draft, W.A.; Writing—review & editing, Y.W. and R.A.; Supervision, W.Z.; Funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 61976103, No. 62272191), the Science and Technology Development Program of Jilin Province (No. 20220201153GX).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data is unavailable due to privacy.

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

References

1. Miranda, A.; Jacobo, E. Extracting a causal network of news topics. In Proceedings of the OTM International Conferences "On the Move to Meaningful Internet Systems", Rome, Italy, 10–14 September 2012; pp. 33–42.
2. Khoo, C.; Kornfilt, J. Automatic extraction of cause-effect information from newspaper text without knowledge-based inferencing. *Lit. Linguist. Comput.* **1998**, *13*, 177–186. [CrossRef]
3. Girju, R. Automatic Detection of Causal Relations for Question Answering. In Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering, Sapporo, Japan, 11 July 2003; Volume 12, pp. 76–83.
4. Luo, Z.; Sha, Y.; Zhu, K.Q.; Wang, Z. Commonsense Causal Reasoning between Short Texts. In Proceedings of the Fifteenth International Conference on Principles of Knowledge Representation and Reasoning, KR'16, Cape Town, South Africa, 25–29 April 2016; pp. 421–430.
5. Gordon, A.S.; Bejan, A.; Sagae, K. Commonsense Causal Reasoning Using Millions of Personal Stories. In Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 7–11 August 2011.

6. Radinsky, K.; Davidovich, S.; Markovitch, S. Learning causality for news events prediction. In Proceedings of the 21st International Conference on World Wide Web, Lyon, France, 16–20 April 2012; pp. 909–918.
7. Silverstein, C.; Brin, S.; Motwani, R.; Ullman, J. Scalable techniques for mining causal structures. *Data Min. Knowl. Discov.* **2000**, *4*, 163–192. [CrossRef]
8. Oh, J.H.; Torisawa, K.; Kruengkrai, C.; Iida, R.; Kloetzer, J. 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, Cambridge, UK, 6–10 February 2017; pp. 415–424.
9. Araúz, P.L.; Faber, P. Causality in the Specialized Domain of the Environment. In Proceedings of the Semantic Relations-II. Enhancing Resources and Applications Workshop Programme, Istanbul, Turkey, 22 May 2012; p. 10.
10. Sachs, K.; Perez, O.; Pe'er, D. Causal protein-signaling networks derived from multiparameter single-cell data. *Science* **2005**, *308*, 523–529. [CrossRef] [PubMed]
11. Wolff, P. Representing causation. *J. Exp. Psychol.* **2007**, *136*, 82–111. [CrossRef] [PubMed]
12. Wolff Phillip; Song Grace. Models of causation and the semantics of causal verbs. *Cogn. Psychol.* **2003**, *47*, 276–332. [CrossRef] [PubMed]
13. Hobbs, J.R. Toward a Useful Concept of Causality for Lexical Semantics. *J. Semant.* **2005**, *22*, 181–209. [CrossRef]
14. Talmy, L. *Toward a Cognitive Semantics. Concept Structuring Systems*; MIT Press: Cambridge, MA, USA, 2000; Volume 1, pp. 1–565.
15. Khoo, C.; Chan, S. Extracting causal knowledge from a medical database using graphical patterns. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, China, 3–6 October 2000; pp. 336–343.
16. White, P.A. Ideas about causation in philosophy and psychology. *Psychol. Bull.* **1990**, *108*, 3–18. [CrossRef]
17. Khoo, C.; Chan, S.; Niu, Y. The Many Facets of the Cause-Effect Relation. In *The Semantics of Relationships*; Part of the Information Science and Knowledge Management Book Series; Green, R., Bean, C.A., Myaeng, S.H., Eds.; Springer: Dordrecht, The Netherlands, 2002; pp. 51–70.
18. Theodorson, G.; Theodorson, A. *A Modern Dictionary of Sociology*; Crowell: New York, NY, USA, 1969; 469p.
19. Pearl, J. Causal inference in statistics: An overview. *Stat. Surv.* **2009**, *3*, 96–146. [CrossRef]
20. Hassanzadeh, O.; Bhattacharjya, D.; Feblowitz, M. Answering Binary Causal Questions Through Large-Scale Text Mining: An Evaluation Using Cause-Effect Pairs from Human Experts. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; pp. 5003–5009.
21. Bozsahin, H.C.; Findler, N.V. Memory-Based Hypothesis Formation: Heuristic Learning of Commonsense Causal Relations from Text. *Cogn. Sci.* **1992**, *16*, 431–454. [CrossRef]
22. Hidey, C.; Mckeown, K. Identifying Causal Relations Using Parallel Wikipedia Articles. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Volume 1, pp. 1424–1433.
23. Asghar, N. Automatic Extraction of Causal Relations from Natural Language Texts: A Comprehensive Survey. *arXiv* **2016**, arXiv:1605.07895. [CrossRef]
24. Bethard, S.; Martin, J.H. Learning Semantic Links from a Corpus of Parallel Temporal and Causal Relations. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers, Columbus, OH, USA, 16–17 June 2008; pp. 177–180.
25. Yang, X.; Mao, K. Multi level causal relation identification using extended features. *Expert Syst. Appl.* **2014**, *41*, 7171–7181. [CrossRef]
26. Li, P.; Mao, K. Knowledge-oriented Convolutional Neural Network for Causal Relation Extraction from Natural Language Texts. *Expert Syst. Appl.* **2019**, *115*, 512–523. [CrossRef]
27. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv* **2014**, arXiv:1406.1078.
28. LeCun, Y.; Neural, Y.B. Convolutional networks for images, speech, and time series. In *The Handbook of Brain Theory and Neural Networks*; MIT Press: Cambridge, MA, USA, 1995; Volume 3361, p. 1995.
29. Kruengkrai, C.; Torisawa, K.; Hashimoto, C.; Kloetzer, J.; Oh, J.-H.; Tanaka, M. Improving Event Causality Recognition with Multiple Background Knowledge Sources Using Multi-Column Convolutional Neural Networks. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31, pp. 3466–3473.
30. Vaswani, A.; Brain, G.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
31. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K.; Language, G.A.I. 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, Minneapolis, MN, USA, 2–7 June 2017; Volume 2, pp. 4171–4186.
32. Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F. TinyBERT: Distilling BERT for natural language understanding. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16–20 November 2020; pp. 4163–4174.
33. Zamri, N.E.; Azhar, S.A.; Mansor, M.A.; Alway, A.; Kasihmuddin, M.S.M. Weighted Random k Satisfiability for k = 1,2 (r2SAT) in Discrete Hopfield Neural Network. *Appl. Soft Comput.* **2022**, *126*, 109312. [CrossRef]

34. Guo, Y.; Kasihmuddin, M.S.M.; Gao, Y.; Mansor, M.A.; Wahab, H.A.; Zamri, N.E.; Chen, J. YRAN2SAT: A novel flexible random satisfiability logical rule in discrete hopfield neural network. *Adv. Eng. Softw.* **2022**, *171*, 103169. [CrossRef]
35. Sidik, S.M.; Mathematics, N.Z. Non-Systematic Weighted Satisfiability in Discrete Hopfield Neural Network Using Binary Artificial Bee Colony Optimization. *Mathematics* **2022**, *10*, 1129. [CrossRef]
36. Nguyen, T.H.; Grishman, R. Relation Extraction: Perspective from Convolutional Neural Networks. In Proceedings of the NAACL-HLT 2015, Denver, CO, USA, 31 May–5 June 2015; pp. 39–48.
37. Turian, J.; Ratinov, L.; Bengio, Y. Word representations: A simple and general method for semi-supervised learning. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 11–16 July 2010; pp. 384–394.
38. Hashimoto, C.; Torisawa, K.; Kloetzer, J.; Sano, M. Toward future scenario generation: Extracting event causality exploiting semantic relation, coantext, and association features. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 23–24 June 2014; pp. 987–997.
39. Zhao, S.; Liu, T.; Zhao, S.; Chen, Y.; Nie, J.-Y. Event causality extraction based on connectives analysis. *Neurocomputing* **2016**, *173*, 1943–1950. [CrossRef]
40. Li, Z.; Ding, X.; Liu, T.; Hu, J.E.; Durme, B. Van Guided Generation of Cause and Effect. *arXiv* **2020**, arXiv:2107.09846.
41. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
42. Kalchbrenner, N.; Grefenstette, E.; Blunsom, P. A Convolutional Neural Network for Modelling Sentences. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June 2014; pp. 655–665.
43. Kim, Y. Convolutional Neural Networks for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
44. Yin, W.; Schütze, H. Convolutional neural network for paraphrase identification. In Proceedings of the NAACL HLT 2015—2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–5 June 2015.
45. Dos Santos, C.N.; Xiang, B.; Zhou, B. Classifying relations by ranking with Convolutional neural networks. In Proceedings of the 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, Beijing, China, 27–31 July 2015; Volume 1, pp. 626–634.
46. Dong, L.; Wei, F.; Zhou, M.; Xu, K. Question answering over freebase with multi-column convolutional neural networks. In Proceedings of the 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, Beijing, China, 27–31 July 2015; Volume 1.
47. Zeng, D.; Liu, K.; Chen, Y.; Zhao, J. Distant supervision for relation extraction via Piecewise Convolutional Neural Networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015.
48. Ciresan, D.; Meier, U.; Schmidhuber, J. Multi-column Deep Neural Networks for Image Classification. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3642–3649.
49. Ponti, E.M.; Korhonen, A. Event-related features in feedforward neural networks contribute to identifying causal relations in discourse. In Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-Level Semantics, Valencia, Spain, 3 April 2017; pp. 25–30.
50. Kayesh, H.; Islam, M.S.; Wang, J. On Event Causality Detection in Tweets. *arXiv* **2019**, arXiv:1901.03526. [CrossRef]
51. Nauta, M.; Bucur, D.; Seifert, C. Causal Discovery with Attention-Based Convolutional Neural Networks. In Proceedings of the Machine Learning and Knowledge Extraction, Canterbury, UK, 26–29 August 2019; Volume 1, pp. 312–340.
52. Zhao, K.; Ji, D.; He, F.; Liu, Y.; Ren, Y. Document-level event causality identification via graph inference mechanism. *Inf. Sci.* **2021**, *561*, 115–129. [CrossRef]
53. Li, Z.; Li, Q.; Zou, X.; Ren, J. Causality extraction based on self-attentive BiLSTM-CRF with transferred embeddings. *Neurocomputing* **2021**, *423*, 207–219. [CrossRef]
54. Liang, S.; Zuo, W.; Shi, Z.; Wang, S. A Multi-level Neural Network for Implicit Causality Detection in Web Texts. *Neurocomputing* **2022**, *481*, 121–132. [CrossRef]
55. Khetan, V.; Rizvi, M.I.; Huber, J.; Bartusiak, P.; Sacaleanu, B.; Fano, A. MIMICause: Representation and automatic extraction of causal relation types from clinical notes. In *Findings of the Association for Computational Linguis*; Association for Computational Linguistics: Dublin, Ireland, 2022; pp. 764–773. [CrossRef]
56. Nayak, T.; Sharma, S.; Butala, Y.; Dasgupta, K.; Goyal, P.; Ganguly, N. A Generative Approach for Financial Causality Extraction. In Proceedings of the Companion Proceedings of the Web Conference 2022 (WWW '22 Companion), Virtual Event, Lyon, France, 25–29 April 2022; Volume 1, pp. 24–26.
57. Johnson, R.; Zhang, T. Deep Pyramid Convolutional Neural Networks for Text Categorization. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Long Papers. Vancouver, BC, Canada, 30 July–4 August 2017; Volume 1, pp. 562–570.
58. Prasad, R.; Dinesh, N.; Lee, A.; Miltsakaki, E.; Robaldo, L.; Joshi, A.K.; Webber, B.L. The Penn Discourse TreeBank 2.0. In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco, 28–30 May 2008; pp. 1–8.

59. Santoro, A.; Raposo, D.; Barrett, D.G.T.; Malinowski, M.; Pascanu, R.; Battaglia, P.; Lillicrap, T. A simple neural network module for relational reasoning. In Proceedings of the Advances in Neural Information Processing Systems 30 (2017), Long Beach, CA, USA, 4–9 December 2017; pp. 1–10.
60. Palm, R.B.; Deepmind, U.P.; Winther, O. Recurrent Relational Networks. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, QC, Canada, 2–8 December 2018; pp. 1–11.
61. Pavez, J.; Allende, H.; María, F.S.; Allende-Cid, H. Working Memory Networks: Augmenting Memory Networks with a Relational Reasoning Module. In Proceedings of the ACL 2018, Melbourne, Australia, 15–20 July 2018; pp. 1–10.
62. Poli, R.; Healy, M.; Kameas, A. *Theory and Applications of Ontology: Computer Applications*; Springer: Dordrecht, The Netherlands, 2010.
63. Ruppenhofer, J.; Ellsworth, M.; Petruck, M.R.L.; Johnson, C.R.; Scheffczyk, J. *FrameNet II: Extended Theory and Practice*; International Computer Science Institute: Berkeley, CA, USA, 2016; pp. 1–119.
64. Shi, Y.; Meng, J.; Wang, J.; Lin, H.; Li, Y. A Normalized Encoder-Decoder Model for Abstractive Summarization Using Focal Loss. In *Natural Language Processing and Chinese Computing*; Part of the Lecture Notes in Computer Science Book Series; Springer: Cham, Switzerland, 2018; pp. 383–392. [CrossRef]
65. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *42*, 318–327. [CrossRef]
66. Kingma, D.P.; Lei Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations, San Diego, CA, USA, 7–9 May 2015.