1Cademy @ Causal News Corpus 2022: Enhance Causal Span Detection via Beam-Search-based Position Selector

Xingran Chen[3]*, Ge Zhang[1 2 3]*, Adam Nik[2 4], Mingyu Li[2 3], Jie Fu[†] [1]

[1] Beijing Academy of Artificial Intelligence, China

[2] 1Cademy Community, USA

[3] University of Michigan Ann Arbor, USA

[4] Carleton College, USA

fujie AT baai.ac.cn

Abstract

In this paper, we present our approach and empirical observations for Cause-Effect Signal Span Detection—Subtask 2 of Shared task 3 (Tan et al., 2022a) at CASE 2022. The shared task aims to extract the cause, effect, and signal spans from a given causal sentence. We model the task as a reading comprehension (RC) problem and apply a token-level RC-based span prediction paradigm to the task as the baseline. We explore different training objectives to fine-tune the model, as well as data augmentation (DA) tricks based on the language model (LM) for performance improvement. Additionally, we propose an efficient beam-search post-processing strategy to due with the drawbacks of span detection to obtain a further performance gain. Our approach achieves an average F_1 score of 54.15 and ranks 1^{st} in the CASE competition. Our code is available at https://github.com/Gzhang-umich/1CademyTeamOfCASE.

1 Introduction

Event extraction has long been a challenging and popular area for natural language processing (NLP) researchers. There are known classic benchmarks, including ACE-2005 (Christopher et al., 2005) and ERE (Song et al., 2015). In recent years, more and more interesting corpora about event detection and extraction have emerged based on different specific source corpora, including biomedical literature (Kim et al., 2003), scientific knowledge resources (Jain et al., 2020), Wiki (Li et al., 2021), and trade-related news (Zhou et al., 2021). In sharp contrast, Cause-Effect Signal Span Detection aims to extract the cause, effect, and signal spans from sentences that have cause-effect relations. Cause-Effect Signal Span Detection is an innovative and important event detection/extraction task that assists in understanding causal relationships from comprehensive sentence samples.

As a new corpus with great potential in event extraction challenges, the Causal News Corpus (CNC) (Tan et al., 2022b) contains socio-political event (SPE) text data with annotated causal spans. The CNC event extraction challenge[1] is the first Cause-Effect Signal Span Detection challenge on a social political news corpus. The challenge itself provides a limited number of annotated samples for supervision, making it more difficult compared to other challenging event extraction tasks. The exploration of causality in news data and the detection of corresponding spans is helpful in reading comprehensive language expressions, making CNC attractive to NLP researchers.

In this paper, we describe our RC-based model with a carefully designed post-processing strategy. We also conduct ablation studies to analyze the influence of both different training objectives and different hyper-parameter settings of the post-processing strategy on our model. In addition, we apply an LM-based data augmentation strategy to further better performance gains, given the low-resource challenge. Our approach improves performance by a large margin in Cause-Effect Signal Span Detection compared to any other competitors.

The main contributions of our paper are as follow:

- We propose an RC-based model with an original post-processing strategy.
- We achieve state-of-the-art performance on the new Cause-Effect Signal Span Detection competition on the CNC.
- We apply an LM-based data augmentation technique to the challenge and prove its positive effect on the challenge of low resources.

[*] The two authors contributed equally to this work.

[†] Corresponding Author

[1] https://github.com/tanfiona/CausalNewsCorpus

Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE), pages 100 - 105
December 7-8, 2022 ©2022 Association for Computational Linguistics

Table 1: Dataset statistics. Avg. Signal represents the average number of Signal spans in each split of dataset.

	Train	Valid	Test	Total
# Sentences	160	15	89	264
# Relations	183	18	119	320
Avg. Signal	0.67	0.56	0.82	0.72

2 Causal News Corpus

The corpus we used in our model training and evaluation is the CNC dataset (Tan et al., 2022b). This dataset is built on the extraction of social-political events from News (AESPEN) (Hürriyetoğlu et al., 2020) in 2020 and the CASE 2021 workshop @ ACL-IJCNLP (Hürriyetoğlu et al., 2021). Each sample in the dataset is annotated with causal labels, that is, whether a sentence contains a causal event. Furthermore, some sentences are annotated with the span of the specific Cause and Effect of a causal event, as well as the signal markers that imply the causality. The spans are labeled by <ARG0>, <ARG1>, and <SIG> annotations to represent the cause, effect, and causal signal in the sentence, respectively. Note that it is possible to have multiple annotations for the same sentence in the dataset if the sentence contains multiple casual relationships of events. The dataset statistics are shown in Table 1.

3 Methodology

In this section, we describe in detail the methodology we used in the task. To begin, we introduce the baseline model established from a pre-trained language model for the task. Next, a beam-search-based post-processing method is introduced to solve the overlap span detection problem in the baseline model. To address the problem that not all examples have signal markers within the sentence, we propose training a signal classifier to determine whether we need to find the signal span of the target test sample. Finally, a pre-trained paraphrasing model is applied for data augmentation.

3.1 Baseline

Algorithm 1 beam-search-based span selector

```
Input: $P_{s_c}, P_{e_c}, P_{s_{ef}}, P_{e_{ef}}, n, k, m$.
Output: $H = \{(s_1, e_1, s_2, e_2, t_i = CBeforeE/CAfterE) : i \leq m\}$
CBeforeE = $\{p^i_{s_c} + p^j_{e_{ef}} : 1 \leq i, j \leq n\}$.
CAfterE = $\{p^i_{s_{ef}} + p^j_{e_c} : 1 \leq i, j \leq n\}$.
Find position pairs with Top-$k$ largest score from both CBeforeE and CAfterE.
Denote the gotten position pairs set as $PS = \{(sp_i, ep_i, t_i = CBeforeE/CAfterE) : sp_i \leq ep_i\}$. $t_i$ implies whether the pair is retrieved from CBeforeE or CAfterE.
Initialize a min heap $H$.
for $ps_p = (sp_p, ep_p, t_p)$ in $PS$ do
    if $t_p$ = $CBeforeE$ then
        Find the position pair $(i, j)$ with the largest $p^i_{e_c} + p^j_{s_{ef}}$, which satisfies $sp_p \leq i \leq j \leq ep_p$.
        Calculate $sc_{(sp_p,i,j,ep_p)} = p^{sp_p}_{s_c} + p^i_{e_c} + p^j_{s_{ef}} + p^{ep_p}_{e_{ef}}$.
    else
        Find the position pair $(i, j)$ with the largest $p^i_{e_{ef}} + p^j_{s_c}$, which satisfies $sp_p \leq i \leq j \leq ep_p$.
        Calculate $sc_{(sp_p,i,j,ep_p)} = p^{sp_p}_{s_{ef}} + p^i_{e_{ef}} + p^j_{s_c} + p^{ep_p}_{e_c}$.
    Push $\{(sp_p, i, j, ep_p), t_p, sc_{(sp_p,i,j,ep_p)}\}$ into $H$.
    if $len(H) > m$ then
        $heappop(H)$ based on $sc_{(sp_p,i,j,ep_p)}$.
return $H$
```

To solve the task, we first fine-tune the pre-trained language model based on the reading comprehension training fashion proposed by BERT (Devlin et al., 2019). Specifically, assume that we need to predict a span within sentence $x = \{t_1, ..., t_n\}$, where t_i is the i^{th} token of sentence x. We can obtain a contextualized representation h_i of t_i using the pre-trained language model:

$$H = \{h_1, ..., h_n\} = BERT(x) \quad (1)$$

Next, we define two parameterized vectors: $v_s, v_e \in R^d$ to calculate the probability that the i^{th} token is the start / end position:

$$P_s = \{p_s^{(1)}, ..., p_s^{(n)}\} = Softmax(v_s^T H) \quad (2)$$

$$P_e = \{p_e^{(1)}, ..., p_e^{(n)}\} = Softmax(v_e^T H) \quad (3)$$

We select the positions with maximum probability as the prediction of the model:

$$s = \underset{1 \leq i \leq n}{\operatorname{argmax}} p_s^{(i)}, \quad (4)$$

$$e = \underset{1 \leq j \leq n}{\operatorname{argmax}} p_e^{(j)}, \quad (5)$$

where s, e represent the predicted start/end position, respectively.

The prediction of the spans of cause, effect, and signal are all similar to the span prediction task described above. For convenience, we will denote the start/end position of cause, effect, and signal as $s_c, e_c, s_{ef}, e_{ef}, s_{sig}, e_{sig}$, respectively, to specify which span we are detecting. Therefore, the training objective is to maximize the probability of ground-truth positions in the model.

3.2 Beam-search-based Span Selector

The proposed baseline model has two drawbacks. First, it is possible that the end position is right before the start position. Second, it is possible to generate spans that overlap each other, which is not allowed in the challenge. Thus, we need to introduce constraints in post-processing to ensure that: 1) the predicted end position must be after the start position of the same span, and 2) the predicted spans of cause and effect do not overlap with each other. In this sub-section, we describe our modified beam search-based algorithm to address the overlapping issue. The beam search algorithm is widely used to find the most possible output with tractable memory and time usage in text generation tasks (Xie, 2017). In reading comprehension or question answering, it is also used to introduce constraint information (Hu et al., 2019), and therefore encourage more accurate predictions. Given a paragraph with length n, we can calculate $P_{s_c} = \{p_{s_c}^{(1)}, ..., p_{s_c}^{(n)}\}$ based on the process introduced in § 3.1. Similarly, we can calculate P_{e_c}, $P_{s_{ef}}$, and $P_{e_{ef}}$ accordingly. Formally, given the input probability vectors P_{s_c}, P_{e_c}, $P_{s_{ef}}$, $P_{e_{ef}}$, a hyper-parameter m denoting the requested answer number, and a hyper-parameter k denoting the beam search size, the span selector is expected to output position pairs s_c, e_c, s_{ef} and e_{ef}. We describe the span selector in detail in Algorithm 1. We denote the proposed span selector as **BSS**. It should be noted that the proposed BSS post-processing algorithm can also generate multiple predictions for cases containing multiple causal relations. For example, we could change the hyperparameter m to retrieve the prediction of cause/effect spans combinations with the top-m highest scores as our predictions of multiple causal relations. For the signal span, we always use the span with the highest score as our prediction (if it presents).

3.3 Signal Classifier

We observe that some samples do not have signal markers (spans) within the sentence even while the baseline model predicts s_{sig}, e_{sig} for each target sample. Therefore, we propose to train a classifier to address this issue. Specifically, we first automatically annotate training samples based on whether signal markers appear within the samples. Then, we fine-tune the pre-trained language model to train a binary classifier. Note that we can share the language model parameters between signal classifier and span detection, i.e. we optimize both training objectives during our fine-tuning process. In addition, we can also train a signal classifier with a separate language model. In our experiments, we apply the two methods separately and compare their effectiveness.

3.4 Data Augmentation with Pre-trained Paraphrasing Model

Considering that only 183 training samples are available for subtask 2, it is important to introduce the data augmentation trick to increase the size of the training dataset. Therefore, in this work, we propose using language models to paraphrase the existing data. Specifically, we use a PEGASUS model (Zhang et al., 2020) fine-tuned for paraphrasing [2] to re-write the phrases of Cause, Effect in each sample. For example, for a training sample "*<ARG1>The farmworkers ' strike resumed on Tuesday</ARG1> when <ARG0>their demands were not met</ARG0>.*", we paraphrase the cause and effect spans within the sample, then obtain the augmented sample "*<ARG1>On Tuesday, the farmworkers resumed their strike</ARG1> when <ARG0>their demands weren't met</ARG0>.*". In this case, the semantic meaning of the original sentence is preserved. Hence, the annotation of the original sample is still reasonable and can continue to be used in the augmented sample. In our implementation, n new phrases were generated for each span. Namely that each sample will end up with n^2 augmented samples. We denote the trick as **DA**.

4 Experiments

In this section, we present the experimental details of training the model and discuss the performance of our proposed approach.

4.1 Experimental Details

In our experiment, we use Albert (Lan et al., 2019) as our LM backbone. We perform hyper-parameter searching to find the best hyper-parameter setting. Specifically, we select the learning rate l

[2] We directly use fine-tuned checkpoint in https://huggingface.co/tuner007/pegasus_paraphrase

Table 2: Experimental results and related ablation study on subtask 2. The evaluation metric of all the results is F_1. Note that n represents the hyper-parameter of data augmentation described in § 3.4.

Methods	**Cause**	**Effect**	**Signal**	**Overall**
Baseline	77.8	66.7	53.5	68.2
Baseline-NER	57.8	57.4	10.8	47.4
Baseline + DA ($n = 2$)	72.2	77.8	60.9.	71.9
Baseline + BSS + DA ($n = 2$)	77.8	**83.3**	60.9	74.1
Baseline + ES + DA ($n = 2$)	72.2	77.8	76.7	75.4
Baseline + JS + DA ($n = 2$)	72.2	72.2	71.3	69.8
Baseline + BSS + ES + DA ($n = 2$)	77.8	**83.3**	76.7	77.5
Baseline + BSS + ES + DA ($n = 3$)	**83.3**	77.8	**80.0**	**80.4**

from $\{1e-5, 2e-5, 5e-5\}$, batch size b from $\{1, 2, 4, 8, 16, 32\}$. We fine-tune the pre-trained model for 30 epochs, and select the checkpoint with the best performance on the development set to conduct evaluation on the test set. Our implementation is based on `Huggingface` (Wolf et al., 2019).

In terms of the signal classifier, we consider two settings: 1) We fine-tune the signal classifier in conjunction with the main training objective as described in § 3.3. We denote this approach as **Joint Sig. (JS)**; 2) We additionally fine-tune a language model to specifically decide whether to predict the span of Signal. We denote this approach by **Extra Sig. (ES)**

We also include another implementation of the baseline recommended by the organizers, where the fine-tuning process is carried out in the end-to-end fashion of Named Entity Recognition (NER). We denote this baseline by **Baseline-NER**.

4.2 Main Results and Ablation Study

Here, we present and discuss the experimental results of our best-performing method for this task, together with the corresponding ablation study. Note that all results are evaluated on the dev set, due to the inaccessibility of the test dataset. We present the score of different approaches F_1 on all three span detection in Table 2.

The results clearly show that the reading comprehension style of the training significantly improves the effectiveness of the approach. We can also observe that it is better to apply the reading comprehension training fashion than token-level tagging for the causal span detection task. Regarding our proposed approaches, the LM-based paraphrasing data augmentation technique improves the performance of the approach by a large margin compared to the baseline. The improvement is consistent, that is, there is an improvement in the prediction of all types of spans. In addition, our proposed BSS post-processing algorithm further improves our approach. However, it can be seen that the improvement of the approach by BSS mainly comes from the prediction of cause and effect. This is reasonable because the algorithm does not post-process the predictions of Signal. As for the signal classifier, both ES and JS make an improvement, which comes mainly from the better prediction of Signal. However, note that the improvement in ES is larger. We conjecture that it might be because of a new training objective introduced by JS, which is harmful to the proposed approach to learning to predict the spans better. Finally, we mix all of the approaches together with our approach and ended up with the best performance. Here, we also compared the impact of data augmentation at different scales. Specifically, we compare the results when $n = 2$ (4× dataset size) with $n = 3$ (9× dataset size). We find that higher data augmentation sizes lead to better results in the validation dataset.

4.3 Case Study of Data Augmentation

In this subsection, we provide a case study on the effectiveness of data augmentation proposed in the system. The comparisons between generated texts and the original texts are shown in Table 3.

From the results, the expressions in the data-augmented texts are more diverse while remaining semantically consistent with the original sentence. Furthermore, the data-augmented texts are competitive with the original in terms of fluency and grammatical correctness.

Table 3: Case Study of Data Augmentation. Note that we generate two sentences for Cause and Effect, respectively. Therefore, there are in total 4 outcomes sentences via combinations.

Ori.	<ARG1>The farmworkers ’ strike resumed on Tuesday</ARG1>when <ARG0>their demands were not met</ARG0>
DA	<ARG1>On Tuesday, the farmworkers resumed their strike</ARG1>when <ARG0>their demands weren’t met</ARG0>. <ARG1>On Tuesday, the farmworkers resumed their strike</ARG1>when <ARG0>their demands didn’t get met</ARG0>. <ARG1>On Tuesday, the farmworkers went on strike</ARG1>when <ARG0>their demands weren’t met</ARG0>. <ARG1>On Tuesday, the farmworkers went on strike</ARG1>when <ARG0>their demands didn’t get met</ARG0>.

Table 4: Overall performance of the proposed approach on the test set. The numbers in parentheses represent the rankings.

Final Competition Results	
Recall	0.5387 (1)
Precision	0.5509 (2)
F1	0.5415 (1)
Accuracy	0.4315 (1)

4.4 Competition Result

We reveal and discuss the final results of our proposed approach competition on a test set. The results are shown in Table 4.

As shown in the table, our proposed approach achieves state-of-the-art results in 3 out of 4 evaluation metrics on subtask 2. This shows the excellent performance of the proposed approach in solving the task of causal spans detection.

5 Conclusion

This paper introduces a reading comprehension-based method, an original post-processing strategy, and an LM-based data augmentation trick for the new Cause-Effect Signal Span Detection competition. We compare the RC-based method with the NER-based one and prove that the RC-based method gets an observing performance gain compared to the NER-based one. We provide experimental results and ablation studies of our beam-search-based Span Selector and LM-based data augmentation tricks to analyze their efficiency and prove their compatibility with other tricks. Our approach achieves state-of-the-art performance in the new competition.

Acknowledgements

This study was supported by the National Key R&D Program of China (2020AAA0105200).

References

Walker Christopher, Strassel Stephanie, Medero Julie, and Maeda Kazuaki. 2005. Ace 2005 multilingual training corpus.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Minghao Hu, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. *arXiv preprint arXiv:1908.05514*.

Ali Hürriyetoğlu, Hristo Tanev, Vanni Zavarella, Jakub Piskorski, Reyyan Yeniterzi, Osman Mutlu, Deniz Yuret, and Aline Villavicencio. 2021. Challenges and applications of automated extraction of socio-political events from text (CASE 2021): Workshop and shared task report. In *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, pages 1–9, Online. Association for Computational Linguistics.

Ali Hürriyetoğlu, Vanni Zavarella, Hristo Tanev, Erdem Yörük, Ali Safaya, and Osman Mutlu. 2020. Automated extraction of socio-political events from news (AESPEN): Workshop and shared task report. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, pages 1–6, Marseille, France. European Language Resources Association (ELRA).

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

J-D Kim, Tomoko Ohta, Yuka Tateisi, and Jun’ichi Tsujii. 2003. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut.

2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online. Association for Computational Linguistics.

Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. 2015. From light to rich ERE: Annotation of entities, relations, and events. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, Denver, Colorado. Association for Computational Linguistics.

Fiona Anting Tan, Hansi Hettiarachchi, Ali Hürriyetoğlu, Tommaso Caselli, Onur Uca, Farhana Ferdousi Liza, and Nelleke Oostdijk. 2022a. Event causality identification with causal news corpus - shared task 3, CASE 2022. In *Proceedings of the 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2022)*, Online. Association for Computational Linguistics.

Fiona Anting Tan, Ali Hürriyetoğlu, Tommaso Caselli, Nelleke Oostdijk, Tadashi Nomoto, Hansi Hettiarachchi, Iqra Ameer, Onur Uca, Farhana Ferdousi Liza, and Tiancheng Hu. 2022b. The causal news corpus: Annotating causal relations in event sentences from news.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Ziang Xie. 2017. Neural text generation: A practical guide. *arXiv preprint arXiv:1711.09534*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. Trade the event: Corporate events detection for news-based event-driven trading. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.