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

Abstract

In this paper, we present our approach and empirical observations for Cause-Effect Signal Span Detection—Subtask 2 of Shared task 3[1] 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 F1 score of 54.15 and ranks 1st in the CASE competition. Our code is available at https://github.com/Gzhang-umich/1CademyTeamOfCASE.

Causal News Corpus Data

The corpus we used in our model training and evaluation is the CNC dataset[2]. 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.

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

Methodology

We first 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.

We approach the task of causality span detection as a **Reading Comprehension (RC)** task and use a **Beam-Search Post-Processing Strategy** to correct for overlap between the cause and effect spans

Beam Search Algorithm

Given the input probability vectors P_{sc} , P_{ec} , P_{sef}, P_{eef}, where $p_{sc}^{(i)}$ is the probability of that the i[th] token of the sentence is the *start* of the *cause* span, 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 the token positions for *sc*, *ec*, *sef* and *eef* . We describe the span selector in detail in Algorithm 1. We denote the proposed span selector as BSS. For the signal span, we always use the span with the highest score as our prediction (if it presents).

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

Experiment Set Up

In our experiment, we use Albert[3] as our LM backbone. We perform hyper-parameter searching to find the best hyper-parameter setting. Specifically, we select the learning rate *l* 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*[4]. In terms of the signal classifier, we consider two settings: 1) we fine-tune the signal classifier in conjunction with the main training objective, denoted as **Joint Sig. (JS)** and 2) we fine-tune an additional language model to specifically decide whether to predict the span of Signal, denoted as **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.

Main Results

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**

Competition Results

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.

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)

References

1.Fiona Anting Tan, Ali Hürriyetog̀lu, Tommaso Caselli, Nelleke Oostdijk, Hansi Hettiarachchi, Tadashi Nomoto, Onur Uca, and Farhana Ferdousi Liza. 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.

2 .Fiona Anting Tan, Ali Hürriyetog̀lu, Tommaso Caselli, Nelleke Oostdijk, Tadashi Nomoto, Hansi He tiarachchi, Iqra Ameer, Onur Uca, Farhana Ferdousi Liza, and Tiancheng Hu. 2022b. The causal news corpus: Annotating causal relations in event sentences from news. In *Proceedings of the Language Resources and Evaluation Conference*, pages 2298– 2310, Marseille, France. European Language Resources Association.

3. 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.*

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