LIORI at the FinCausal 2021 Shared task: Transformer ensembles are not enough to win

Adis Davletov
RANEPA, Moscow, Russia
Lomonosov Moscow State University, Moscow, Russia
davletov-aa@ranepa.ru

Sergey Pletenev
RANEPA, Moscow, Russia
HSE University, Moscow, Russia
pletenev-sa@ranepa.ru

Denis Gordeev
RANEPA, Moscow, Russia
Russian Foreign Trade Academy, Moscow, Russia
gordeev-di@ranepa.ru

Abstract

In this paper, we report on our system for FINCAUSAL 2021 Financial Document Causality Detection Task. In this task, the aim is to identify, in a causal sentence or text block, the causal elements and the consequential ones. We propose a system that uses a pre-trained model, fine-tuned on the extended dataset, and task-specific post-processing of the model's inputs to improve the quality of the results. We tried two types of approaches: 1) a fine-tuned T5-model that generated cause and effect spans 2) and a sequence-to-sequence model based on XLNet that solved the task as token classification. The best result of our XLNet-large is 0.946 F1 on the test set while T5-model got the F1 score of 0.835 which may be due to the lower number of exact matches.

1 Introduction

Causality detection is an important problem as a vital part of natural language understanding. It is especially true for the domain of finance and economics where causes should contribute to the prediction model while effects should either be used as an output or omitted from the model altogether. A major contribution to the field was provided in the workshop FinCausal 2020 (Mariko et al., 2020) where the authors have provided a labelled dataset for causality and effect detection and a platform for the discussion of the results and further aligned measurement of the models. It contained two tracks: the first task was to classify whether a sentence contains causality or not, while the second one was devoted to the extraction of causes and effects from the sentences.

This work is focused on our approach to FinCausal 2021 [1]. Unlike the last year competition, this year shared task consisted of a single track equivalent to the second track from 2020. The dataset consists of texts each containing causes and effects. One text may contain several spans of the same type.

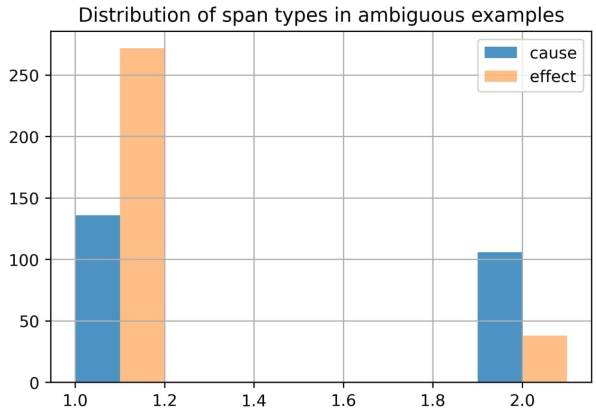

Figure 1: Ambiguous text examples

The winning solution of the 2020 2nd subtask consisted in a BERT-CRF system with a Viterbi decoder (Kao et al., 2020). This year we also tried to implement their solution but were unsuccessful with this approach and got the F1 score of 0.875 on our test dataset derived from 2021 training data.

2 Preprocessing

In this work we experimented on two datasets:

- FinCausal-2021 dataset, which consists of two subtasks, including causal meanings detection (Task 1) and cause-effect detection (Task 2). The numbers of training instances are 22,058

[1] http://wp.lancs.ac.uk/cfie/fincausal2021/

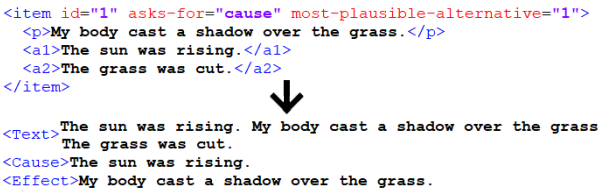

Figure 2: Example of Balanced-COPA dataset transform

and 1,750 for Task 1 and Task 2, respectively. This work only focuses on Task 2.

- Balanced-COPA dataset (Kavumba et al., 2019), The Choice Of Plausible Alternatives dataset (COPA) contains 1000 examples of two question types: a) What is the CAUSE of this? and b) What is the EFFECT of this?. Balanced COPA contains one additional, mirrored instance for each original training instance in COPA. A total of 1,500 examples are given.

Since we are experimenting with two different models, the data sets will be handled differently for both models.

For the Seq2Seq model, the data set is split into two parts. The model is to predict the cause part and the effect part of a text in two steps. In the first step, the model predicts only the Cause part of the text, is the second step – only the Effect part. To control which part of the sentence the model should generate, we use additional prefixes in the text that will be given to the input of the model. At the end of each text, we prepend an additional prefix as shown in 3.

For Sequence Tagging models we used encode_causal_tokens function provided by organizers to reformat all data to BIO format.

3 Models

The task of extracting textual relations can eventually be generalized to the task of selecting a subset of the words or sentences in the text. There are many approaches to solving this task. We have selected two variants: Sequence tagging and Sequence-to-Sequence generation.

3.1 Sequence tagging

In case of sequence tagging models we experimented with BERT(Devlin et al., 2018), RoBERTa(Liu et al., 2019), ALBERT(Lan et al., 2019) and XLNet(Yang et al., 2019) models. Models were asked to predict sequences of labels in BIO format. Every input example was tokenized using 'word_tokenize' function from the NLTK library (Bird et al., 2009). Optionally, we feed the information about the number of input sentences to the models, which we get using PunktTrainer and PunktSentenceTokenizer from NLTK library trained on all provided textual data. We concatenate the one-hot encoded number of sentences to the output from pretrained models. Our models are quite similar to the BERT-base model from (Kao et al., 2020). The differences are in our post-processing steps and different training scheme. Also, we experiment with linear or non-linear classifier layers over pretrained models. In post-processing steps, we apply four transformations similar to the rules worked out by the Workshop organizers to annotate the training data (Mariko et al., 2020). They are:

Text: The politician lost the election. No one voted for him. He ran positive campaign ads.

—

Input: The politician lost the election. No one voted for him. He ran positive campaign ads. Question: cause

Output: No one voted for him

Input: The politician lost the election. No one voted for him. He ran positive campaign ads. Question: effect

Output: The politician lost the election

Figure 3: Example of additional prefixes for text

- If a sentence contained only one fact (cause or effect), we tagged the entire sentence.
- The annotation of sentence-to-sentence causal relationships is prioritized
- When a causal chain is located inside a single sentence, in order to facilitate the extraction process, we chose to span the causal units as much as possible
- If two facts of the same type were located in the same sentence and were related to the same effect or cause, then we annotated these two facts as one unit

Rules	F1	Exact Match
Basic model	86.24	69.58
Rule 1	86.32	69.73
Rule 12	87.25	71.14
Rule 123	86.04	38.53
Rule 1234	86.21	38.53
Rule 124	**87.44**	**71.76**

Table 1: Results for rule combinations on the dev set for the best performing model

We trained our models for 15 epochs making validation every quarter of epoch saving the best models. We tested all models with all sequences of rules and chose the rule combination with the best F1 score and Exact Match. It appeared to be the rule combination 1-2-4 (see Table 1). As our final submitted model, we use a voting ensemble of two ALBERT models and one XLNet model.

3.2 Sequence-to-Sequence

Modern Sequence-to-Sequence models are successful in many tasks. In this paper we use the T5 model (Raffel et al., 2020). The T5 model is trained on several data sets for 18 different tasks, which are split into 8 categories: summarizing text, question answering, translation, etc.

We use HuggingFace Transformers [2] for T5 training and prediction. The model is trained with the following parameters: encoder length 512, decoder length 256, batch size 2, 8 epochs, learning rate 5e-05, after every 1000 steps we evaluate our models with beam size 8.

To get different results in multi-effect or multi-cause cases we use the diverse beam-search(Vijayakumar et al., 2018): If the resulting hypothesis from diverse beam-search starts with a different symbol, they are presented as new results.

4 Results

Model	F1
Viterbi (Kao et al., 2020)	0.875
BARTNER (Yan et al., 2021)	0.7729
T5_large	0.868
T5_large_2	0.8741

Table 2: Viterbi-BERT and T5 analysis on our development set

[2]https://huggingface.co/transformers/

Model	F1
T5 Sequence-to-Sequence	0.835267
ALBERT XXLarge	0.93984
XLNet-base-cased	0.925649
ensemble of 2 ALBERT XXLarge and 1 XLNet models	0.946473

Table 3: Results on the evaluation dataset

The final results of our models are shown in Table 3. As can be seen from the table, our Sequence tagging models outperform T5 Sequence prediction models.

We have also tested the 2020 Fincausal winning solution (Kao et al., 2020) and BARTNER (Yan et al., 2021) as an alternative to T5. BARTNER also solves the token classification problem as a sequence classification task. T5 outperformed BARTNER, while its performance was close to BERT+Viterbi. However, we failed to replicate the results for the Viterbi model. It might be due to hyperparameter tuning or some mistake in processing. All in all, our token classification turned out to be the most successful among the ones that we have tried.

XLNet and ALBERT models have been trained using 2 Nvidia GeForce RTX 2080 TI GPUs. T5 models have been trained using GPUs provided by Google Colab Pro.

5 Error Analysis

Such a low result in T5 compared to regular token tagging is probably due to the fact that the model sometimes has difficulty in predicting perfectly all the tokens in the input text. While token tagging only works with the source text, the t5 model may mix up or miss some tokens due to its seq2seq nature. This is confirmed by the low exact match result in the table.

6 Conclusion

In this work, we describe our results for the FinCausal 2021 dataset. We have tried Sequence Classification and Sequence-to-Sequence models. Sequence classification outperformed Sequence-to-Sequence in our case. We have also tested various sequence post-processing schemes and ensembles of Transformer-based models.

Acknowledgements

We would like to thank the organisers for the task and the workshop.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Pei-Wei Kao, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2020. NTUNLPL at FinCausal 2020, task 2:improving causality detection using Viterbi decoder. In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 69–73, Barcelona, Spain (Online). COLING.

Pride Kavumba, Naoya Inoue, Benjamin Heinzerling, Keshav Singh, Paul Reisert, and Kentaro Inui. 2019. When choosing plausible alternatives, clever hans can be clever.

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. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Dominique Mariko, Hanna Abi-Akl, Estelle Labidurie, Stephane Durfort, Hugues De Mazancourt, and Mahmoud El-Haj. 2020. The financial document causality detection shared task (FinCausal 2020). In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 23–32, Barcelona, Spain (Online). COLING.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R. Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search: Decoding diverse solutions from neural sequence models.

Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various ner subtasks. *arXiv preprint arXiv:2106.01223*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.