DSC-IITISM at FinCausal 2021: Combining POS tagging with Attention-based Contextual Representations for Identifying Causal Relationships in Financial Documents

Gunjan Haldar[1⊕], **Aman Mittal**[1⊕], **Pradyumna Gupta**[2⊕]
[1]Department of Mechanical Engineering
[2]Department of Electronics Engineering
[⊕]Indian Institute of Technology (ISM), Dhanbad 826004, India

Abstract

Causality detection draws plenty of attention in the field of Natural Language Processing and linguistics research. It has essential applications in information retrieval, event prediction, question answering, financial analysis, and market research. In this study, we explore several methods to identify and extract cause-effect pairs in financial documents using transformers. For this purpose, we propose an approach that combines POS tagging with the BIO scheme, which can be integrated with modern transformer models to address this challenge of identifying causality in a given text. Our best methodology achieves an F1-Score of 0.9551, and an Exact Match Score of 0.8777 on the blind test in the FinCausal-2021 Shared Task at the FinCausal 2021 Workshop.

1 Introduction

Integrating causality information as text features can substantially benefit a plethora of applications such as text mining (Girju and Moldovan, 2002), event prediction (Wei and Wang, 2019), question answering (Sharp et al., 2016) and many more. One of the primary motives of this, which has been explored in this challenge, is to extract causality in the financial domain, which can be applied to various tasks such as financial services support (Chen et al., 2020), consumer review (Patil, 2016), stock movement prediction (Chen, 2021) as well as help different institutions to gain insights into the financial sector.

By examining the financial documents carefully, one can observe that single and multiple causal events in a given paragraph may exist. Additionally, there can also be the existence of numerous causal chains in the same. To deal with such cases, we formulate causality detection and extraction task as a sequence labeling and modeling problem and propose an approach using POS tagging (Dhumal Deshmukh and Kiwelekar, 2020) with BIO scheme tagging (Liu et al., 2015) integrated with an ensemble of BERT Large-cased (Devlin et al., 2018), XLNet Base (Yang et al., 2019), BERT Large-Cased Whole Word Masking, GPT-2 (Radford et al., 2019) and RoBERTa Base (Liu et al., 2019), achieving an F1-Score of 0.9551 and Exact Match score of 0.8777 on Blind test dataset provided by the workshop.

2 Dataset

The dataset provided (Mariko et al., 2021) (Mariko et al., 2020) for this challenge[1] has been extracted from 2019 financial documents provided by Qwam[2], consisting of the complete text and the extracted cause and effect pairs along with offset markers. It was also observed that multiple instances comprised of the same text but different causality pairs, due to presence of multiple chains of causal relationships. Total instances present in the database were 2393 which were split into 2101 training and 292 validation instances.

3 Methodology

3.1 Part-Of-Speech (POS) Tagging

We tokenize each sentence and generate rule-based part-of-speech (POS) tags (Dhumal Deshmukh and Kiwelekar, 2020) for each token. Rule-based POS tagging uses contextual information and a set of handwritten rules to assign POS tags to tokens in a sentence.

After tokenizing the data, the tokens are converted into POS tags. The POS tags are enumerated, which are further mapped on the tokenized sentences. These POS tags are represented in the form of a one-hot vector. This vector is concatenated with the model's hidden state output of the last layer, which is then sent to the final linear layer

[1] http://wp.lancs.ac.uk/cfie/fincausal2021/
[2] https://www.qwamci.com/

Token	POS Tag	BIO Tag	Token	POS Tag	BIO Tag
The	DT	B-E	It	PRP	B-C
Sunshine	NNP	I-E	is	VBZ	I-C
State	NNP	I-E	consistently	RB	I-C
drew	VBD	I-E	one	CD	I-C
...	...	I-E	...	...	I-C
...	...	I-E	...	...	I-C
older	JJR	I-E	taxes	NNS	I-C

Table 1: Pre-processed Output stored in text format, The above text represents an example instance from the training set.

of the model. Predictions are performed on the concatenated vector or tensor.

Tag	Description	BIO Label
B-E	At the Beginning of Effect	3
B-C	At the Beginning of Cause	1
I-C	Inside of Cause	2
I-E	Inside of Effect	4
-	Padding	0

Table 2: Tagging Scheme explanation. BIO tag "O" will be converted to padding.

3.2 BIO Scheme Tagging

To extract the causal relations and positional information of the words, considering the semantics of the causal events, we use the BIO tagging (Liu et al., 2015) scheme i.e. Begin-Inside-Outside tagging with Cause and Effect labels (C-E). BIO tagging scheme will represent whether the token is at the beginning (B) of the target phrase, inside (I) of a target phrase and tokens which are not a part of cause or effect are considered as being outside (O) of the target phrase and are labelled as padding (-). Additionally, due to varying sequence length, extra tokens which are not included in cause and effect tuples are converted to padding as shown in Table 2.

3.3 Pre-processing

To begin with, two different modes are given as input for pre-processing. When the mode is "training", the corresponding sentence and cause-effect tuples in the training data are append to a dictionary, otherwise when the mode is "test", sentences in the test dataset are appended to a dictionary. Each sentence in the paragraph is tokenized, subsequently, separate tokens and their positional index are stored in a list.

Further, for the preparation of BIO tags, the index of the tokenized words are identified in each sentence using its respective index and stored in a dictionary. The beginning of cause and effect pairs are found in the sentence, and this pair is tokenized. Tokens at the beginning of the cause and effect are labelled as B-C and B-E respectively. Subsequent tokens in cause and effect sentences are labelled as I-C and I-E respectively. These labels along with the words are stored in a dictionary identified by their index. The tags are extracted from the dictionary. This process is iterated over all the instances in the training set.

To end with, each word is concatenated with its respective POS tags and BIO tags as shown in Table 1. The pre-processed file is stored in a text format which is further passed onto the model as input.

3.4 Transformer Architecture

For the purpose of this challenge, our best approach utilizes an ensemble developed using BERT (Bidirectional Encoder Representations from Transformers) Large-Cased model (Devlin et al., 2018), RoBERTa (Robustly Optimized BERT Pre-training Approach) (Liu et al., 2019), GPT-2 (Generative Pre-trained Transformer) (Radford et al., 2019), BERT Large-Cased Whole Word Masking (Devlin et al., 2018) (BWM), XLNet (Yang et al., 2019) by applying the huggingface[3] (Wolf et al., 2019) package.

3.4.1 Models

BERT Large-cased transformer model has been pre-trained on the English language with a masked language modeling (MLM) objective distributed into Masked Language Modelling and Next Sentence Prediction (NSP), which converges to learn

[3] https://huggingface.co/bert-large-cased

Model	Epochs	MSL*	Validation Score†	Blind Test†
BERT-base	40	256	0.9197	0.9253
RoBERTa-base	50	256	0.9201	0.9372
GPT-2	20	128	0.9251	0.9422
XLNet-base	50	128	0.9368	0.9466
BERT-large	**50**	**256**	**0.9389**	**0.9517**
BWM	50	256	0.9327	0.9476

Table 3: Model Comparison by Experimentation; *Maximum Sequence Length, †F1 Score

Model	F1-Score	Recall	Precision	Exact Match
BERT-large + RoBERTa + XLNet + GPT-2 + BWM	0.9551	0.9580	0.9554	0.8777

Table 4: Best performing method on the official Blind Test of FinCausal-2021

an internal representation that can be utilized to extract features from downstream tasks. This model consists of 24 transformer encoder layers with 1024 hidden dimensions with 16 self-attention heads.

BWM model has been pre-trained on the same language corpus as BERT Large-Cased model but with a whole word masking technique, wherein all of the tokens corresponding to a word are masked at once. The overall masking rate remains the same. The model was pre-trained on 4 cloud TPUs for one million steps with a batch size of 256. The sequence length was limited to 128 tokens for 90% of the steps and 512 for the remaining 10%. The optimizer used is Adam with a learning rate of 1e-4, β_1 = 0.9 and β_2 = 0.999 and a weight decay of 0.01

RoBERTa is pre-trained with the same objective as BERT but on 1024 V100 GPUs for 500K steps with a batch size of 8K and a sequence length of 512. Adam optimizer is used with a learning rate of 6e-4, β_1 = 0.9, β_2 = 0.98 and ϵ = 1e-6, and a weight decay of 0.01 with dynamic masking where the model randomly masks 15% of the words in the input then run the entire masked sentence through the model and has to predict the masked words.

GPT-2 transformer model takes sequences of continuous text as input and uses an internal mask-mechanism to predict the token at any position "i" by the inputs at position 1 to "i".

XLNet is a generalized autoregressive pre-training method enabling learning of bidirectional contexts.

3.4.2 Training

The pre-processed output file was procured, and for every instance, the corresponding POS and BIO tags of each token was extracted and stored in an array. According to the maximum sequence length, these arrays were padded. Depending on the transformer model utilized, `[CLS]` and `[SEP]` tokens were appended to the tokens. For instance, if the transformer model was BERT Large-Cased, `[CLS]` token was appended at the beginning and `[SEP]` token at the end and when the transformer model is XLNet, `[CLS]` is appended at the end. Pseudo POS tag ID and BIO tag ID for `[CLS]` or `[SEP]` token was set as "0" and "-100" respectively. All ID sequences were padded with padding token ID - "0" in POS tag sequence and "-100" in BIO tag sequence.

Each model consumed on an average 3-4 hours for training. The configurations of the best models which are used for the ensemble are reported in Table 3. All these models have been trained with a batch size of 64 with cross-entropy loss (Gordon-Rodriguez et al., 2020) so that only real IDs contribute to the loss function and not the padding IDs.

3.5 Post-processing & Exact Match Optimization

The received predictions are in the format of tuples of tokens and their corresponding predicted BIO tag. The BIO tags are retrieved and stored in a list with the index of each token in the prediction. Further, this process is iterated over all the predicted instances and recorded. We tried to optimize the Exact Match metric by selecting the longest cause-effect pair when multiple causal chains are present in a given data instance. If the number of padding tokens was less than a given threshold between two similar predicted phrases (Cause/Effect), the two pairs were merged.

Text	Cause	Effect
The company also recently announced a quarterly dividend, which was paid on Tuesday, September 3rd. Shareholders of record on Thursday, August 15th were paid a $0.03 dividend. This represents a $0.12 annualized dividend and a yield of 3.42%.	The company also recently announced a quarterly dividend, which was paid on Tuesday, September 3rd.	Shareholders of record on Thursday, August 15th were paid a $0.03 dividend.
	The company also recently announced a quarterly dividend, which was paid on Tuesday, September 3rd.	This represents a $0.12 annualized dividend and a yield of 3.42%.
If you pay the full RAD there is no interest (DAP) pay no RAD and you will pay a DAP which is the interest on the full amount: $22,160.	pay no RAD	you will pay a DAP which is the interest on the full amount: $22,160.
	If you pay the full RAD	there is no interest (DAP)

Table 5: Table representing identical multi-causal chains. Causal chains in the training dataset.

3.6 Ensemble

After each prediction was extracted from different models present in the ensemble, the mode was calculated to find the most frequently occurring label. In the presence of a tie-breaker scenario, we select the label predicted by the best performing single transformer model, BERT Large-Cased. Further, after extracting all the tags, these were aligned with the text to get the actual words bundled together to form the cause-effect pair.

4 Experimentation and Results

Different models along with custom loss functions were trained on the given data and local F1 score, Recall, and Precision were evaluated. Transformer models including RoBERTa (Robustly Optimized BERT Pre-training Approach) (Liu et al., 2019), GPT-2 (Generative Pre-trained Transformer) (Radford et al., 2019), BERT Base (Devlin et al., 2018), BERT Large-Cased Whole Word Masking (Devlin et al., 2018) (BWM), XLNet (Yang et al., 2019) were experimented with different hyper-parameter settings. The best performing settings along with their corresponding scores are reported in Table 3. The results were evaluated locally, and considering those metrics, the model performance was observed. To boost up optimization, ensembles of the aforementioned transformer models were experimented and evaluated.

GPT-2 was trained and experimented with, but due to expensive computational requirements, it was trained for 20 epochs. Loss function while RoBERTa-base transformer model was being trained on the data couldn't converge, resulting in a low metric score; similar behavior was observed in XLNet. BERT large-cased model outperformed all these models due to its large architectural layout when a single shot transformer is concerned. Maximum Sequence Length (MSL) is a critical factor while training a model with limited computational resources, because having a high MSL means most of the memory is wasted for padding and not used for weight update. Subsequently, smaller MSL values are chosen for transformer models with vast architecture. Ensembles mentioned in Table 4 gave a relatively low F1 score when BERT-base was included along with other models indicating that the lower performance of BERT-base single shot experiment could be the prominent dropping factor. The performance metrics of the top approach is shown in Table 4.

5 Conclusion

This paper presents our sequence labeling and modeling approach, combining POS tags with BIO scheme using ensemble optimization strategy comprising BERT-large, RoBERTa, XLNet, GPT-2, and BERT-Large (whole word masking) for causality detection in financial documents which helped us achieve the highest Exact Match score of 0.8777, on the FinCausal-2021 Shared Task leaderboard. Future works can describe an optimization pipeline constituting architecturally larger transformer models. Furthermore, more advanced post-processing strategies can be investigated to extract multiple causal relationships in a text.

References

Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2020. Nlp in fintech applications: Past, present and future. *ArXiv*, abs/2005.01320.

Qinkai Chen. 2021. Stock movement prediction with financial news using contextualized embedding from bert.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Rushali Dhumal Deshmukh and Arvind Kiwelekar. 2020. Deep learning techniques for part of speech tagging by natural language processing. In *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, pages 76–81.

Roxana Girju and Dan Moldovan. 2002. Text mining for causal relations.

Elliott Gordon-Rodriguez, Gabriel Loaiza-Ganem, Geoff Pleiss, and John P. Cunningham. 2020. Uses and abuses of the cross-entropy loss: Case studies in modern deep learning.

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 *FNP*.

Shengyu Liu, Buzhou Tang, Qingcai Chen, and Xiaolong Wang. 2015. Drug name recognition: Approaches and resources. *Information*, 6:790–810.

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. *CoRR*, abs/1907.11692.

Dominique Mariko, Hanna Abi Akl, Estelle Labidurie, Hugues de Mazancourt, and Mahmoud El-Haj. 2021. The Financial Document Causality Detection Shared Task (FinCausal 2021). In *The Third Financial Narrative Processing Workshop (FNP 2021)*, Lancaster, UK.

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 *The 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation (FNP-FNS 2020)*, Barcelona, Spain.

Mayur Patil. 2016. Summarization of customer reviews for a product on a website using natural language processing.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Rebecca Sharp, Mihai Surdeanu, Peter Jansen, Peter Clark, and Michael Hammond. 2016. Creating causal embeddings for question answering with minimal supervision. *CoRR*, abs/1609.08097.

Xiang Wei and Bang Wang. 2019. A survey of event extraction from text. *IEEE Access*, PP:1–1.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.