

CYUT at the NTCIR-16 FinNum-3 Task: Data Resampling and Data Augmentation by Generation

Xie-Sheng Hong, Jia-Jun Lee, Shih-Hung Wu

Chaoyang University of Technology
Taichung, Taiwan

Mike Tian-Jian Jiang

Zeals Co, Ltd
Tokyo, Japan

Outline

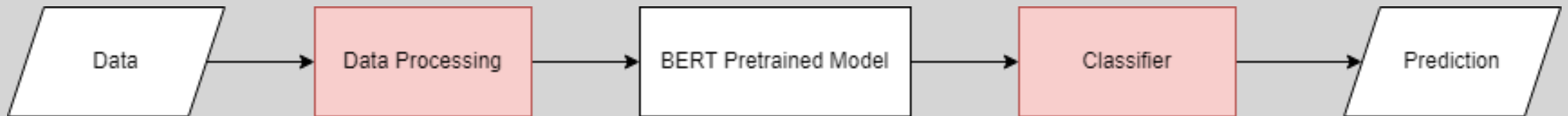
- Task Define and Difficulty
- Method: Deep Learning and BERT
- Proposed Runs
 - CYUT-1
 - CYUT-2
 - CYUT-3
- Model Configurations
- Additional Runs
- Official Runs and Additional Runs
- Discussion
- Conclusions and Future Work

Task Define and Difficulty

- Task Define
 - To predict whether a number is a "claim" in a given description
 - A binary classification task
- Difficulty
 - Training dataset imbalance.
 - Analyst's Report (Chinese data): 1 : 0.3
 - Earnings Conference call (English): 1 : 0.14
 - **Model trained by this may bias towards specific label**

Method: Deep Learning and BERT

- Deep Learning:
 - Using deep learning method to solve the problem
- BERT(Bidirectional Encoder Representations from Transformers)[1]:
 - Pretrained Model
 - Fine-tune and transfer learning
 - Using variant of BERT



Proposed Runs

- CYUT-1: MacBERT / RoBERTa with BiLSTM
 - Baseline for our all systems
 - Data resampling
- CYUT-2: AWD-LSTM
 - Based on CYUT-1.
 - Replace BiLSTM with AWD-LSTM
 - Data resampling
- CYUT-3: Additional data
 - Based on CYUT-1.
 - GPT-2 generates additional data

CYUT-1 : BiLSTM and Data Resampling

- Pretrained Model
 - Analyst's Report (Chinese data): MacBERT[6]
 - Earnings Conference Call (English data): RoBERTa[7]
- The classical classifier
 - BiLSTM
- Data Resampling
 - Repeatedly extract data from lesser number of labels(which is 1).
 - Adjust the data ratio of the 2 labels to 1 : 1

CYUT-2: Replace BiLSTM with AWD-LSTM

- AWD-LSTM(ASGD Weight-Dropped LSTM)[5]
 - It is a variant of LSTM, which is a weight-dropped LSTM
 - It drop part of data of weight matrix between the hidden states in the LSTM
- Advantages
 - 1. It prevents the overfitting problem of traditional LSTM
 - 2. It minimizes the effect on the training speed
- Data Resampling

CYUT-3: GPT-2 Data Generation

- **Not to use data resampling**
 - Using GPT-2[4] generate additional data that label must be 1 (in-claim)
- GPT-2 Model used:
 - Analyst's Report (Chinese data): CLUECorpusSmall, trained by CLUECorpus2020 dataset
 - Dataset: news, wiki, comments from Amazon, etc.
 - Earnings Conference call (English data): Original GPT-2 model without additional training

CYUT-3: GPT-2 Data Generation

- A very intuitive way to generate text with GPT-2:
 - A fixed text + A random number input GPT-2 => Generate subsequent text
- Example:
 - Analyst's Report (Chinese data):
 - Input: 我們推測會上升 X (We predict an increase of X)
 - Output(Fixed length is 100): 我們推測會上升 $X\%$ ，明天早晨大跌...
(We predict an increase of $X\%$ and a big fall tomorrow morning...)
 - Earnings Conference call (English data):
 - Input: ***We anticipate a X increase***
 - Output(Fixed length is 50): We anticipate a X increase in the number of cases with...
 - *Notice: X meaning a random number (range: 0 - 1000)*

CYUT-3: GPT-2 Data Generation

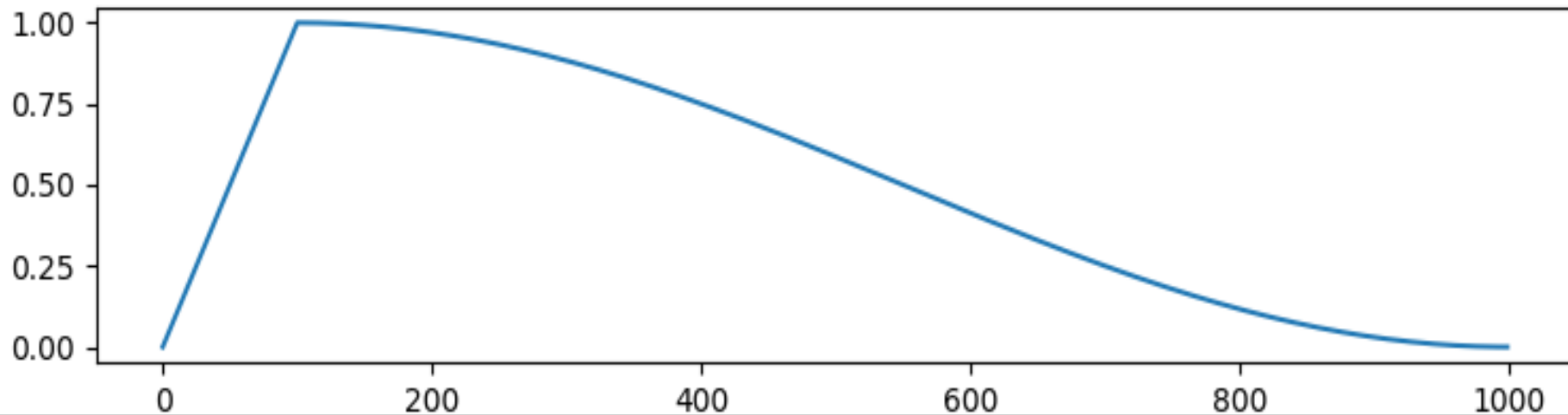
- Amount and rate of data generation
 - Analyst's Report (Chinese data):
 - Amount of generation: 2200
 - Amount of data: 999: 3220 to 3199 : 3220
 - Data rate: 1 : 0.3 to about 1 : 1
 - Earnings Conference call (English data):
 - Amount of generation: 4000
 - Amount of data: 1039 : 7298 to 5039 : 7298
 - Data rate: 1 : 0.14 to about 1 : 0.7

Model Configurations

Parameters	Values
BERT Model	macbert-base or Roberta-base
Batch Size	4 or 8
Max Length	512
Optimizer	AdamW
Learning Rate	2e-5

Model Configurations: Learning Rate Schedule

- Cosine schedule is better linear schedule for our model in this task.
 - Learning rate warmup
 - Dynamically adjusted learning rate



Ref. https://huggingface.co/docs/transformers/main_classes/optimizer_schedules#transformers.get_cosine_schedule_with_warmup.num_cycles

Additional Runs

- Additional run is mainly modified based on CYUT-3:
 - Add the text type used for GPT-2 text generation
 - Adjust the amount of additional text generated
 - Increase or decrease the amount of additional text, etc.
 - No change
 - Only BERT + BiLSTM, not use data augmentation method including data resampling at all

Official Runs and Additional Runs

Analyst's Report			
Run	Macro-F1	Micro-F1	Recall
CYUT-2	86.76%	91.73%	90.32%
CYUT-3	88.20%	92.16%	88.76%
CYUT-1	88.80%	92.11%	87.34%
No Change	88.75%	92.52%	89.30%
More Data and Seeds	89.23%	92.86%	89.92%
More Seeds	89.30%	93.14%	91.66%
1000 Data	89.97%	93.16%	89.52%
More Data	90.24%	93.43%	90.31%

Official Runs and Additional Runs

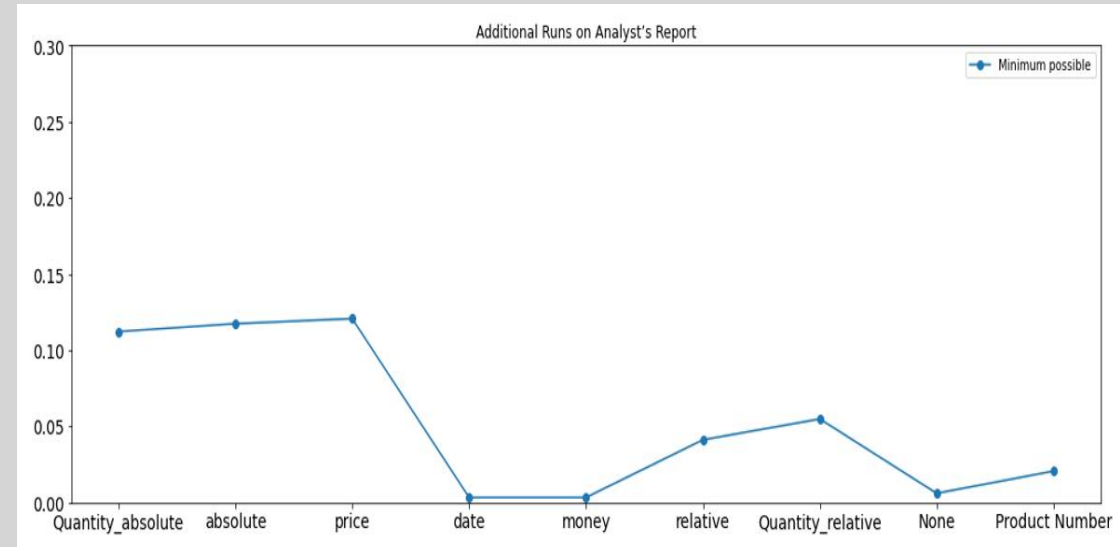
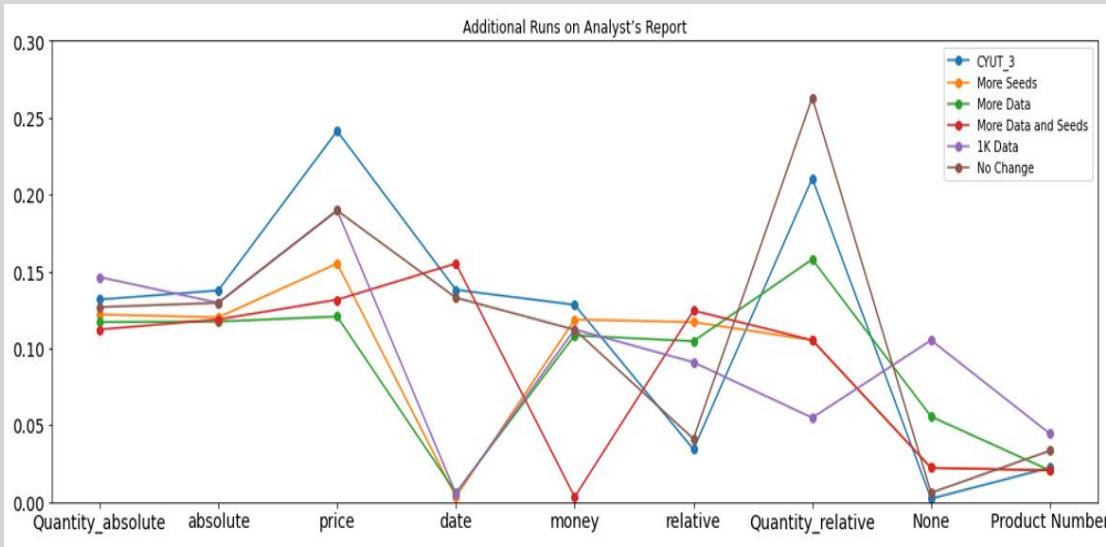
Earnings Conference Call			
Run	Macro-F1	Micro-F1	Recall
CYUT-1	85.53%	94.67%	79.82%
More Seeds	85.93%	95.00%	80.74%
More Data	86.73%	95.93%	84.78%
More Data and Seeds	87.17%	95.76%	83.25%
1000 Data	87.28%	95.73%	83.15%
CYUT-2	87.49%	95.64%	82.39%
CYUT-3	87.88%	96.43%	87.25%
No Change	88.15%	96.22%	85.03%

Discussion

- We Found:
 - Raising the number of additional texts may improve the model, but is not an absolute factor
 - “No Change” is the best system in Earnings Conference Call data
- We assume that:
 - The quality of additional texts may be more important than the amount
 - Poor data quality can be counterproductive
 - Whether data Resampling is causing overfitting?

Discussion

- There are advantages and disadvantages among systems in each category
 - Build a large multi-model system that leverages the strengths of each system



Conclusions and Future Work

- Data augmentation technique for imbalance dataset.
 - Data resampling
 - Data generation
- Official and Additional Run.
 - Analyst's Report (in Macro-F1):
 - Official: 88.80%
 - Additional: 90.24%
 - Earnings Conference Call (in Macro-F1):
 - Official: 87.88%
 - Additional: 88.15%

Conclusions and Future Work

- Pay attention
 - Additional text data quality is more important than amount.
 - Possible overfitting
- Future Work:
 - Improving the quality of data generation
 - T-5 model[2] 、 GPT-3 model[3], etc.
 - Adjusting the BERT model
 - Building multi-model system make the right talent for the right place

Reference

- [1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *arXiv:1810.04805 [cs]*, May 2019, Accessed: Jun. 15, 2021. [Online]. Available: <http://arxiv.org/abs/1810.04805>
- [2] C. Raffel *et al.*, “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” arXiv, arXiv:1910.10683, Jul. 2020. doi: [10.48550/arXiv.1910.10683](https://doi.org/10.48550/arXiv.1910.10683).
- [3] T. B. Brown *et al.*, “Language Models are Few-Shot Learners,” arXiv, arXiv:2005.14165, Jul. 2020. doi: [10.48550/arXiv.2005.14165](https://doi.org/10.48550/arXiv.2005.14165).
- [4] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners,” p. 24.
- [5] S. Merity, N. S. Keskar, and R. Socher, “Regularizing and Optimizing LSTM Language Models,” arXiv, arXiv:1708.02182, Aug. 2017. doi: [10.48550/arXiv.1708.02182](https://doi.org/10.48550/arXiv.1708.02182).
- [6] Y. Cui, W. Che, T. Liu, B. Qin, S. Wang, and G. Hu, “Revisiting Pre-Trained Models for Chinese Natural Language Processing,” *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 657–668, 2020, doi: [10.18653/v1/2020.findings-emnlp.58](https://doi.org/10.18653/v1/2020.findings-emnlp.58).
- [7] Y. Liu *et al.*, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” *arXiv:1907.11692 [cs]*, Jul. 2019, Accessed: Aug. 27, 2021. [Online]. Available: <http://arxiv.org/abs/1907.11692>

Thank You!

Please let me know if you have any questions.