CPIC at SemEval-2023 Task 7: GPT2-based Model for Multi-evidence Natural Language Inference for Clinical Trial Data

Mingtong Huang, Junxiang Ren, Lang Liu, Ruilin Song, Wenbo Yin
China Pacific Insurance (Group) Co. Ltd., China
{huangmingtong, renjunxaing, liulang-009, songruilin, yinwenbo-001}@cpic.com.cn

Abstract

This paper describes our system submitted for SemEval Task 7, Multi-Evidence Natural Language Inference for Clinical Trial Data. The task consists of 2 subtasks. Subtask 1 is to determine the relationships between clinical trial data (CTR) and statements. Subtask 2 is to output a set of supporting facts extracted from the premises with the input of CTR premises and statements. Through experiments, we found that our GPT2-based pre-trained models can obtain good results in Subtask 2. Therefore, we use the GPT2-based pre-trained model to fine-tune Subtask 2. We transform the evidence retrieval task into a binary class task by combining premises and statements as input, and the output is whether the premises and statements match. We obtain a top-5 score in the evaluation phase of Subtask 2.

1 Introduction

In recent years, there has been a significant increase in the number of Clinical Trial Reports (CTR) publications. Currently, the number of CTRs related to breast cancer alone exceeds 10,000. With this increasing trend, it is impossible for clinical practitioners to keep all the existing literature up to date in the future in order to provide personalized, evidence-based care (DeYoung et al., 2020). In this scenario, Natural Language Inference (NLI) (Maccartney, 2009) provides a great opportunity to support the large-scale interpretation and retrieval of medical evidence. A successful natural language inference system enables clinical practitioners to obtain the latest medical evidence for better personalized evidence-based care (Sutton et al.). Therefore, the study of SemEval Task 7 (Jullien et al., 2023) is particularly meaningful to provide technical ideas for the large-scale interpretation and retrieval of medical evidence.

Task 7 is based on CTRs of breast cancer and contains two subtasks. Subtask 1 is Textual Entailment, which aims to determine the relationship (implication and contradiction) between CTRs and statements. Subtask 2 is Evidence Retrieval, which outputs a set of supporting evidence extracted from the CTR premise with given CTR and statements to prove the prediction labels in Subtask 1.

Our system focuses on Subtask 2 which reasons about the statements supported by the facts in the premises by means of the generative capability of GPT-2 (Radford et al., 2019). Firstly, our system concatenates the premises (i.e., facts in CTRs) and the inference(i.e., statements) through the prmopt template to obtain the inference sequence. Then the inference sequence is input into GPT-2, and the output label is 0 or 1. The label 1 means that the premise fact and the statement are successfully matched, so the fact can be added to the corresponding group of supporting facts. Finally, after the matching relationship between each fact and the statement is determined, we can obtain a set of facts supporting the current statement.

Besides, when exploring Subtask 2 with GPT-2, we found that the parameter size of the model has a relatively large impact on the system. At the same time, the fusion between models with different parameters can have a significant improvement on the system. Eventually with these strategies, our system achieved a top-five ranking.

The remainder of this paper is organized as follows: Section 2 is about the task data, task setup, etc. Section 3 shows the overall algorithm and strategy of the system. Section 4 contains the experimental setup, dataset segmentation, etc. Section 5 presents experimental results and analysis. Section 6 is the summary and outlook of the system in this paper.

2 Background

2.1 Task Dataset

The dataset of Task 7 is based on a collection of breast cancer CTRs containing labels annotated by

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 397–401
July 13-14, 2023 ©2023 Association for Computational Linguistics

domain experts based on statements and interpretations. The collected CTRs can be categorized into 4 parts:

1) Eligibility criteria: a set of conditions that allow patients to participate in clinical trials.

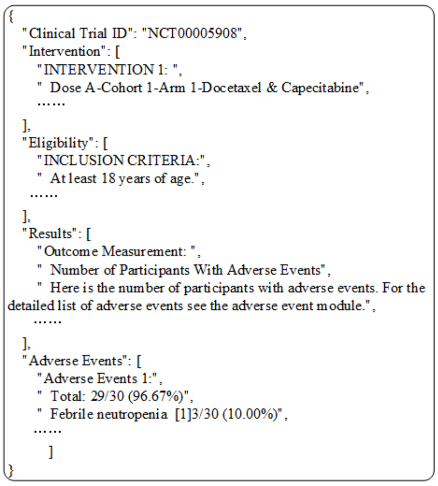
2) Intervention: Information about the type, dose, frequency and duration of the treatment.

3) Results: number of participants, outcome measures, units and results in the trial.

4) Adverse events: signs and symptoms observed during clinical trials.

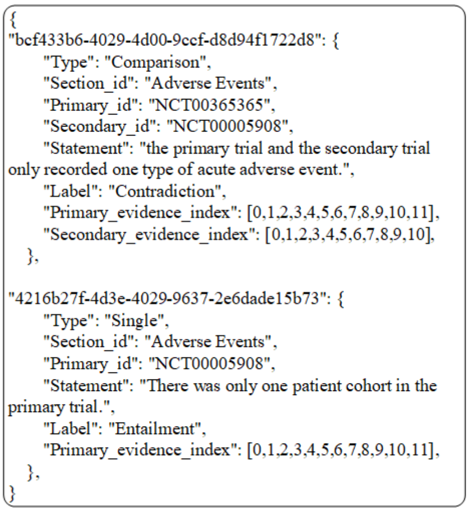
In addition to CTR, there are statements with comments, with an average of 19.5 tokens. A statement is a declaration of some type for one or more of the four parts of the CTR premise. It may make claims for a single CTR, or a comparative declaration for two CTR.

Examples of CTRs and statements are shown in Figure 1 and 2, respecively. Figure 1 shows a CTR, which contains the four parts mentioned above, each consisting of several sentences of facts. Figure 2 is a statement. When the key "Type" is "Comparison", like the first example in Figure 2, the statement is supported by two CTRs. They are used to find a series of facts supporting the statement through the indexes of "Primary_evidence_index" and "Secondary_evidence_index", respectively. When the key "Type" is "Single", like the other example in Figure 2, the statement is supported by a single CTR.



```
{
  "Clinical Trial ID": "NCT00005908",
  "Intervention": [
    "INTERVENTION 1: ",
    "  Dose A-Cohort 1-Arm 1-Docetaxel & Capecitabine",
    ......
  ],
  "Eligibility": [
    "INCLUSION CRITERIA:",
    "  At least 18 years of age.",
   ......
  ],
  "Results": [
    "Outcome Measurement: ",
    "  Number of Participants With Adverse Events",
    "  Here is the number of participants with adverse events. For the
detailed list of adverse events see the adverse event module.",
    ......
  ],
  "Adverse Events": [
    "Adverse Events 1:",
    "  Total: 29/30 (96.67%)",
    "  Febrile neutropenia  [1]3/30 (10.00%)",
    ......
      ]
}
```

Figure 1: Example of CTRs



```
{
"bcf433b6-4029-4d00-9ccf-d8d94f1722d8": {
      "Type": "Comparison",
      "Section_id": "Adverse Events",
      "Primary_id": "NCT00365365",
      "Secondary_id": "NCT00005908",
      "Statement": "the primary trial and the secondary trial
only recorded one type of acute adverse event.",
      "Label": "Contradiction",
      "Primary_evidence_index": [0,1,2,3,4,5,6,7,8,9,10,11],
      "Secondary_evidence_index": [0,1,2,3,4,5,6,7,8,9,10],
   },

"4216b27f-4d3e-4029-9637-2e6dade15b73": {
      "Type": "Single",
      "Section_id": "Adverse Events",
      "Primary_id": "NCT00005908",
      "Statement": "There was only one patient cohort in the
primary trial.",
      "Label": "Entailment",
      "Primary_evidence_index": [0,1,2,3,4,5,6,7,8,9,10,11],
   },
}
```

Figure 2: Example of statements

The input of Subtask 2 is CTRs in Figure 1 and statements in Figure 2, while the output is a group of facts extracted from CTR, like the list value of the key "Primary_evidence_index" in Figure 2.

2.2 Data Analysis

The clinical trial report CTRs and statements are analyzed in Table 1.

Item	Count
Sum of CTR	999
Sum of Statement	2400
Avg length of Statement	19.5
Max length of Statement	65
Avg length of Primary_evidence	10.7
Max length of Primary_evidence	197
Avg length of Secondary_evidence	10.8
Max length of Secondary_evidence	194

Table 1: Analysis of CTRs and statements

We can observe from the table that the length of the text is relatively short. The maximum length of statement is 65 and the average length is 19.5. Besides, the maximum length of "Primary_evidence" in the evidence is 197 and the average length is 10.7, while the maximum length and the average length of "Secondary_evidence" is 194 and 10.8, respectively. The generative model does not take much time to reason in the case of short texts, which motivates us to use the GPT2-based model in the subsequent experiments.

3 System Overview

3.1 Model Structure

GPT (Radford et al., 2018) means generative pre-training. The overall model structure of GPT-2 is the same as GPT. We change the downstream task of GPT-2 to binary classification. Specifically, we connect a linear layer after GPT-2 to do binary classification. Each fact in CTR is connected to the corresponding statement by a template separately and then input to GPT-2 for inference. The overall structure is shown in Figure 3.

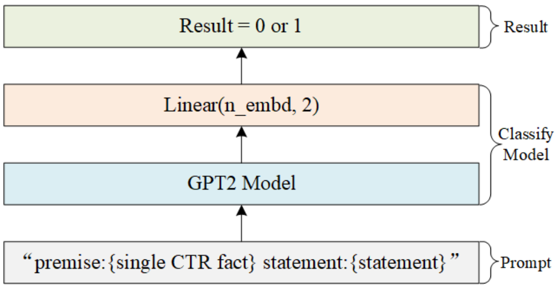


Figure 3: Structure of the model

First, in order to input the CTRs and statements into the GPT-2 model as an inferred relationship, we designed an template, i.e., "premise: statement:". Since the statement is inferred from the facts in CTRs, we concatenate the facts in CTR before the statement. We then fill in the individual fact and the corresponding statement in CTRs according to this order to finally get our sequence input to the subsequent classification model.

Secondly, in the classification model demonstrated in Figure 3, the first layer is the GPT-2 model, which is built by Transformer's Decoder(Vaswani et al., 2017). It has a strong one-way generation capability, which corresponds to the logical order of "premise" to "statement" in our prompt. That is the reason why we use the GPT-2 model. The second layer of the classification model is a linear layer, which compresses the output generated by GPT-2 to 2 dimensions for binary classification.

Finally, we transform the output of the linear layer to get the classification of 0 or 1. 1 means that "premise" and "Statement" are matched. By concatenating each fact in the CTR with the statement separately, we can get all the facts that match the statement in the whole CTR, and combine these facts to get a set of facts that support the statement.

3.2 Cross validation

The model mentioned before yields good results on the initial training and validation sets. To make full use of the training and validation sets we use cross-validation, we merge the training and validation sets together to join the training. The results prove that cross-validation can lead to a remarkable improvement in the online test set. We used a 5-fold cross-validation, and the flow of cross-validation is shown in Figure 4.

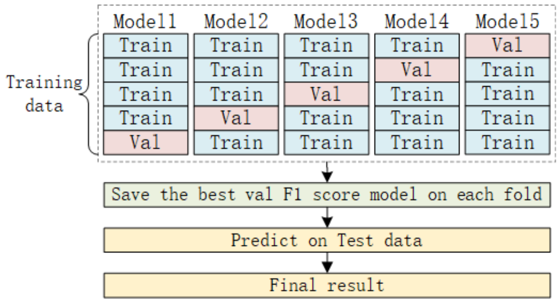

Training data
Model1
Model2
Model3
Model4
Model5
Train Train Train Train Val
Train Train Train Val Train
Train Train Val Train Train
Train Val Train Train Train
Val Train Train Train Train
Save the best val F1 score model on each fold
Predict on Test data
Final result

Figure 4: Cross validation

We split the training data into five equal parts. By setting each of them as a validation set and the remaining parts as a training set, we get five models. Then the five models with the highest F1 values on the validation set are saved separately, which are used to make predictions on the test set and fused to get the final results.

3.3 Model Fusion

The fusion strategy we mainly adopt is to fuse GPT-2 models with different parameter sizes and different random seeds. The final prediction results are determined mainly by voting. The general process is shown in Figure 5.

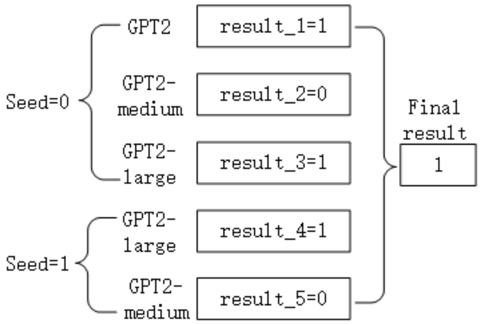


Figure 5: Strategy of Model Fusion

4 Experimental Setup

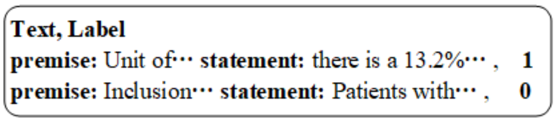
4.1 Data Processing

First of all, the official data are divided into the train set, development set and test set, where the test set is not labeled. The detailed experimental data distribution is shown in Table 2, and the numbers in the table represent the numbers of statement entries.

Data	Train	Dev	Test
Original Size	1700	200	500

Table 2: Size of original data

Next, we need to process the data in Figure 1 and Figure 2 into the format of the prompt in Figure 3 so that they can be fed into the GPT-2 model. If the facts in CTR exist in the list of evidence index of statement, they are labeled as 1. Otherwise, they are labeled as 0. The processed data are shown in Figure 6.



Text, Label
premise: Unit of··· **statement:** there is a 13.2%··· , **1**
premise: Inclusion··· **statement:** Patients with··· , **0**

Figure 6: Format of processed data

The size of the processed data is presented in Table 3. The quantities in the table indicate the numbers of combination of the facts in the CTRs and the corresponding statements.

Data	Train	Dev	Test
Size	39935	4224	10836

Table 3: Size of Processed data

4.2 Implementation Details

Since we convert the subtask into a binary classification task, our initial idea is to use BRET to output the classification results by concatenating the premise and statement with "[SEP]". We began with pre-trained models such as BERT-base (Devlin et al., 2018), ERNIE (Sun et al., 2019), etc. In addition, we also made experiments on T5 model (Raffel et al., 2019) followed by classification linear layer.

In order to make use of the inference from premises to statements in Figure 6, we apply the GPT-2 model. We get the matching result by inputting the whole-sentence text in Figure 6 into three versions of the GPT-2 model, respectively.

During our experiments, we combined the training data and development data in Table 3 and divided them into five parts using cross-validation. To further improve the effect of the model, we fuse the models with different parameter sizes to get the final prediction results.

4.3 Evaluation Metrics

The standard precision, recall and F1-score are used as the evaluation metrics, where the F1-score is the basis for the final ranking.

5 Results

Strategy	F1	P	R
BERT-base	0.719	0.769	0.675
ERNIE	0.744	0.718	0.771
GPT-2 small	0.776	0.740	0.815
T5 base+5folds	0.782	0.792	0.772
GPT-2 small+5folds	0.789	0.771	0.807
GPT-2 med+5folds	0.794	0.796	0.792
GPT-2 large+5folds	0.795	**0.815**	0.775
GPT-2 s+m+l+5folds	**0.810**	0.788	**0.834**

Table 4: Experimental Results

The precision, recall, and F1 in Table 4 are based on the online test set. We can observe from the table that after training the model with the "premise+statement" dataset we construct, the performance of GPT-2-based model is significantly better than that of BERT-based model since GPT-2 can learn the information of premise and statement more effectively.

In terms of the GPT-2-based model, taking GPT-2-small as an example, the addition of 5-fold cross-validation can improve the F1-score by nearly 1%, while the GPT-2 model with larger parameters has better F1-score. Besides, we find that there are obvious differences in the results obtained by models with different parameter sizes. For example, the GPT2-small model has a high recall score on the test set, while GPT2-large has a high precision score on the test set, and GPT2-medium has a relatively balanced precision and recall score.

Therefore, it is natural for us to combine the characteristics of the three and perform model fusion. We find that adjusting the fusion threshold to increase the recall score can improve the overall F1 value more effectively.In addition, the F1-score of GPT2-based model after cross-validation is also higher than that of T5 model. The best result in Table 4 is obtained by fusing two different random seeds, with a total of one small, two medium, and two large models, which improves the F1 value by nearly 1.4% compared to the single best model. The strategy also helps us to achieve the 5th place in the competition.

6 Conclusion

In this paper, we design a system based on GPT2-based model to obtain the matching templates of premise and statement. Our experiment results demonstrate that GPT2-based model significantly outperforms BERT-based model on our constructed data. In addition, the combination of cross-validation and model fusion can lead to significant improvement. Finally, we obtained the top-5 ranking for Subtask 2 of SemEval2023 Task 7. Due to the time constraint of the competition, our model did not take into account the logical information between CTRs, but only matched the individual CTR with the statement, which can be further optimized in the future.

References

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.

Jay DeYoung, Eric Lehman, Benjamin E. Nye, Iain James Marshall, and Byron C. Wallace. 2020. Evidence inference 2.0: More data, better models. *CoRR*, abs/2005.04177.

Mael Jullien, Marco Valentino, Hannah Frost, Paul O'Regan, Donal Landers, and André Freitas. 2023. Semeval-2023 task 7: Multi-evidence natural language inference for clinical trial data. In *Proceedings of the 17th International Workshop on Semantic Evaluation*.

B. Maccartney. 2009. Natural language inference. *STANFORD UNIVERSITY*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.

R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, and K. I. Kroeker. An overview of clinical decision support systems: benefits, risks, and strategies for success. *npj Digital Medicine*.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. Attention is all you need. *arXiv*.