Towards Understanding of Medical Randomized Controlled Trials by Conclusion Generation

Alexander Te-Wei Shieh* **Yung-Sung Chuang*** **Shang-Yu Su** **Yun-Nung Chen**
National Taiwan University, Taipei, Taiwan
{b05401009,b05901033,f05921117}@ntu.edu.tw y.v.chen@ieee.org

Abstract

Randomized controlled trials (RCTs) represent the paramount evidence of clinical medicine. Using machines to interpret the massive amount of RCTs has the potential of aiding clinical decision-making. We propose a RCT conclusion generation task from the PubMed 200k RCT sentence classification dataset to examine the effectiveness of sequence-to-sequence models on understanding RCTs. We first build a pointer-generator baseline model for conclusion generation. Then we fine-tune the state-of-the-art GPT-2 language model, which is pre-trained with general domain data, for this new medical domain task. Both automatic and human evaluation show that our GPT-2 fine-tuned models achieve improved quality and correctness in the generated conclusions compared to the baseline pointer-generator model. Further inspection points out the limitations of this current approach and future directions to explore*.

1 Introduction

Randomized controlled trials (RCTs) are the most rigorous method to assess the effectiveness of treatments, such as surgical procedures and drugs, in clinical medicine (Sibbald and Roland, 1998). A typical RCT often constitutes of two randomized groups of patients receiving either the "intervention" (new treatment) or "control" (conventional treatment). Then, a statistical analysis is done after the experiments to determine whether the intervention has a significant effect (i.e. actually making patients better or worse). The results from various RCTs contribute to the medical decisions made by physicians every day. However, analyzing these large amounts of data could be overwhelming for clinicians (Davidoff and Miglus, 2011). With the help of machine readers, we can alleviate the burden for providing correct information that contributes to critical clinical decisions.

In this work, we aim to evaluate the capabilities of deep learning models on understanding RCTs by generating the conclusions of RCT abstracts. We achieve this by transforming the PubMed 200k RCT abstract sentence classification dataset (Dernoncourt and Lee, 2017) into a RCT conclusion generation task. Generating a *correct* and *coherent* conclusion requires the model to 1) identify the objectives of the trial, 2) understand the result and 3) generate succinct yet comprehensible texts. Therefore, this task can be a preliminary goal toward a more thorough understanding of clinical medicine literature.

To tackle this task, we first build a pointer-generator model (See et al., 2017) as the baseline. This model is widely used in abstractive summarization, which is similar to our conclusion generation task. We then leverage the high quality text generation capability of the Open AI GPT-2 (Radford et al., 2019) language model by fine-tuning the general domain GPT-2 model into a medical domain conclusion generator.

Because the correctness of RCT understanding is essential for supporting clinical decisions and neural summarization models could inaccurately present facts from the source document, we incorporate human evaluation on the correctness and quality of the generated in addition to standard ROUGE score (Lin, 2004) for automated summarization scoring. Evaluation results show the fine-tuned GPT-2 models score higher for both correctness and quality. However, there is still quite a large room for improvement both on the diversity and accuracy of the generated conclusions, providing a guidance for future research directions.

*These authors contribute this paper equally.
*The code is available at: https://github.com/MiuLab/RCT-Gen

Proceedings of the 10th International Workshop on Health Text Mining and Information Analysis (LOUHI 2019), pages 108–117
Hong Kong, November 3, 2019.
https://doi.org/10.18653/v1/D19-62

2 Related Work

The paper focuses on generating RCT conclusions, which is related to natural language generation. We describe the related work below and emphasize the difference between the prior work and our work. In our proposed method, we exploit the state-of-the-art language model representations for understanding the complex medical literature, and related work is then briefly described below.

2.1 Medical Natural Language Generation

Several medical domain natural language generation tasks have been studied using machine learning models, including generating radiology reports from images (Jing et al., 2018; Vaswani et al., 2017) and summarizing clinical reports (Zhang et al., 2018; Pivovarov and Elhadad, 2015) or research literature (Cohan et al., 2018). Recently, Gulden et al. (2019) studied extractive summarization on RCT descriptions.

Abstractive summarization, in which the model directly generates summaries from the source document, is closely related to our conclusion generation task. Most neural approaches for abstractive summarization were based on sequence-to-sequence recurrent neural networks (RNNs) with attention mechanisms (Devlin et al., 2019). The pointer-generator network (See et al., 2017) combined a copy mechanism that directly copies words from the source document and a coverage mechanism to avoid repetition caused by the RNN-based decoder, achieving good performance by handling unseen information. Devlin et al. (2019) further combined intra-encoder and intra-decoder attention with policy learning by using ROUGE-L score as the reward and improved the performance in terms of the evaluation metric. Hsu et al. (2018) combined an extractive model that provided attention on the sentence level and the pointer-generator architecture, and Cohan et al. (2018) also worked on abstractive summarization of long documents, including medical papers from the PubMed database, based on the pointer-generator network.

However, our goal to generate conclusions is different from abstractive summarization in that summarization is to shorten the source document while preserving most of the important information, whereas our conclusion generation model gives one or two sentences describing the main outcome of the given trial. Given the superior performance of pointer-generation networks from the above related summarization work, this paper uses the pointer-generation model as baseline and focuses on RCT conclusion generation instead of abstractive summarization.

2.2 Contextualized Representations

Recent advances of contextualized representation models, such as ELMo (Peters et al., 2018), Open AI GPT (Radford et al., 2018) and BERT (Devlin et al., 2019) achieved remarkable results across different natural language understanding tasks, such as question answering, entailment classification and named entity recognition. At the core of these models was language modeling, with either forward prediction used in GPT, bidirectional prediction used in ELMo, or masked prediction used by BERT. Variants of BERT also improved the performance of bio-medical natural language understanding tasks (Xu et al., 2019; Pugaliya et al., 2019). Peng et al. (2019) further proposed a new benchmark to evaluate the performance of contextualized models in the bio-medical domain.

Particularly, the Open AI GPT-2 model (Radford et al., 2019) has demonstrated rudimentary zero-shot summarization capabilities with only language modeling training. Its forward prediction architecture made it suitable for autoregressive generation in a sequence-to-sequence task. Most benchmarks on contextualized representation were based on sequence classification tasks such as natural language inference and multiple choice question answering (Wang et al., 2018; Peng et al., 2019). Our work, on the other hand, focuses on exploring GPT-2's capability of generating goal-directed sentences in the medical domain. Note that to our knowledge, this paper is the first attempt that investigates GPT-2 towards the medical document understanding and interpretation.

3 Task Formulation

The PubMed 200k RCT dataset was originally constructed for sequential short text classification, with each sentence labeled as "background", "objective", "methods", "results" and "conclusions". We concatenated the "background", "objective" and "results" sections of each RCT paper abstract as the model input and the goal of the model is to generate the "conclusions". Table 1 illustrates

Source: (BACKGROUND) Varenicline is believed to work , in part , by reducing craving responses to smoking cues and by reducing general levels of craving ; however , these hypotheses have never been evaluated with craving assessed in the natural environments of treatment-seeking smokers . (OBJECTIVE) Ecological momentary assessment procedures were used to assess the impact of varenicline on cue-specific and general craving in treatment-seeking smokers prior to quitting . (RESULTS) During all phases , smoking cues elicited greater craving than neutral cues ; the magnitude of this effect declined after the first week . General craving declined across each phase of the study . Relative to the placebo condition , varenicline was associated with a greater decline in general craving over the drug manipulation phase . Varenicline did not significantly attenuate cue-specific craving during any phase of the study .
Target (True Negative): Smoking cues delivered in the natural environment elicited strong craving responses in treatment-seeking smokers , but cue-specific craving was not affected by varenicline administered prior to the quit attempt . These findings suggest that the clinical efficacy of varenicline is not mediated by changes in cue-specific craving during the pre-quit period of treatment-seeking smokers .
Pointer-generator baseline model with $n = 1$ hint word (N/A): smoking cues are associated with a greater craving in general , and may be associated with a greater decline in general craving and
Fine-tuned GPT-2 with $n = 0$ hint word (False Negative): Varenicline did not reduce general craving in treatment-seeking smokers prior to quitting.
Fine-tuned GPT-2 with $n = 1$ hint word (True Negative): Smoking cues are associated with greater general craving than neutral cues, and varenicline does not attenuate cue-specific craving.

Table 1: An example of the GPT-2 $n = 0$ model generating a false negative conclusion (Varenicline did reduce general craving), while the GPT-2 $n = 1$ model generated a better true negative one. The "(BACKGROUND)", "(OBJECTIVE)" and "(RESULTS)" tags denote the sentence classifications according to the original PubMed RCT dataset and are not included in the actual input of our conclusion generation task.

the formulated task, where the generated conclusion needs to contain *correct* information based on the experiments and should be *concise*. After preprocessing, the number of abstracts in the training set is 189,035 and there are 2,479 conclusions used for validation. The average source paragraph length is 170.1 words (6.0 sentences), and the average target conclusion length is 41.4 words (1.8 sentences) long.

4 Models

Language model pre-training has achieved a great success among language understanding tasks with different model architectures. Because training language models requires a large amount of text data, and it is relatively difficult to acquire a lot of RCT documents, this work focuses on first pre-training language models with the transformer architecture (Vaswani et al., 2017) and then adapts the model to support the medical domain by fine-tuning. The language model pre-training from general texts is described below.

4.1 Transformer Encoder in GPT-2

We first introduce the transformer encoder (Vaswani et al., 2017) used as the backbone of the GPT-2 model. The transformer encoder is a stack of N transformer encoder blocks, where the l-th block takes a sequence of hidden representations $X^l = \{X_1^l, \cdots, X_n^l\}$ as the input and outputs an encoded sequence $X^{l+1} = \{X_1^{l+1}, \cdots, X_n^{l+1}\}$.

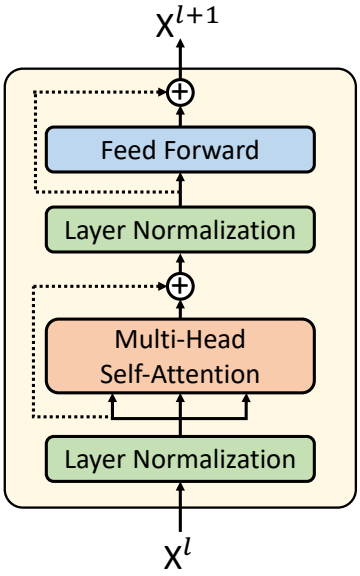


Figure 1: A modified transformer encoder block in the GPT-2 language model.

A transformer encoder block consists of a multi-head self-attention layer and a position-wise fully connected feed-forward layer. A residual connection (He et al., 2016) is employed around each of the two layers followed by layer normalization (Ba et al., 2016). In GPT-2, however, the layer normalization step is moved to the front of the multi-head self-attention layers and the feed-forward layers. An illustration of a GPT-2 transformer encoder block is presented in Figure 1. Each component is briefly described as follows.

Byte-Pair Encoding GPT-2 uses a special byte pair encoding (BPE) for input and output representations. It can cover essentially all Unicode strings, which is useful in processing the medical texts due to the significant out-of-vocabulary problems such as distinct nomenclature and jargon. This special BPE prevents merging characters from different categories and preserves word-level segmentation properties with a space exception.

Positional Encoding Because the transformer model relies on a self-attention mechanism with no recurrence, the model is unaware of the sequential order of inputs. To provide the model with positional information, positional encodings are applied to the input token embeddings

$$X_i^1 = \text{embed}_{\text{token}}[w_i] + \text{embed}_{\text{pos}}[i],$$

where w_i denotes the i-th input token, $\text{embed}_{\text{token}}$ and $\text{embed}_{\text{pos}}$ denote a learned token embedding matrix and a learned positional embedding matrix respectively.

Multi-Head Self-attention An attention function can be described as mapping a query to an output with a set of key-value pairs. The output is a weighted sum of values. We denote queries, keys and values as Q, K and V, respectively. Following the original implementation (Vaswani et al., 2017), a scaled dot-product attention is employed as the attention function. Hence, the output can be calculated as

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V,$$

where d_k denotes the dimension of key vectors.

The idea of multi-head attention is to compute multiple independent attention heads in parallel, and then concatenate the results and project again. The multi-head self-attention in the l-th block can be calculated as

$$\text{MultiHead}(X^l) = \text{Concat}(\text{head}_1, \cdots, \text{head}_h)W^O,$$
$$\text{head}_i = \text{Attention}(X^l W_i^Q, X^l W_i^K, X^l W_i^V),$$

where X^l denotes the input sequence of the l-th block, h denotes the number of heads, W_i^Q, W_i^K, W_i^V and W^O are parameter matrices.

Position-Wise Feed-Forward Layer The second sublayer in a block is a position-wise feed-forward layer, which is applied to each position separately and independently. The output of this layer can be calculated as

$$\text{FFN}(x) = \max(0, x \cdot W_1 + b_1)W_2 + b_2,$$

where W_1 and W_2 are parameter matrices, b_1 and b_2 are parameter biases.

Residual Connection and Layer Normalization As shown in Figure 1, layer normalization is first applied on the input to the multi-head attention and feed-forward sublayers. The residual connection is then added around the two sublayers. The output of the l-th block can be calculated as

$$H^l = \text{MultiHead}(\text{LayerNorm}(X^l)) + X^l,$$
$$X^{l+1} = \text{FFN}(\text{LayerNorm}(H^l)) + H^l.$$

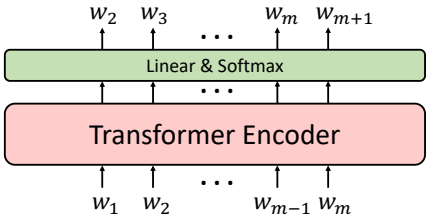
4.2 GPT-2 Pre-Training

The generative pre-training (GPT) via a language model objective is shown to be effective for learning representations that capture syntactic and semantic information without supervision (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019). The GPT model proposed by Radford et al. (2018) employs the transformer encoder with 12 encoder blocks. It is pre-trained on a large generic corpus that covers a wide range of topics. The training objective is to minimize the negative log-likelihood:

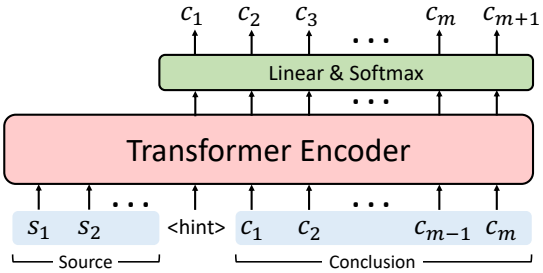
$$\mathcal{L} = \sum_{t=1}^{T} -\log P(w_t \mid w_{<t}, \theta),$$

where w_t denotes the t-th word in the sentence, $w_{<t}$ denotes all words prior to w_t, and θ are parameters of the transformer model.

To avoid seeing the future contexts, a masked self-attention is applied to the encoding process.



(a) The pre-training stage



(b) The fine-tuning stage.

Figure 2: Illustration of the two-stage method in the GPT model. The tag `<hint>` denotes where hint word tokens are introduced during fine-tuning.

In the masked self-attention, the attention function is modified into

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}} + M)V,$$

where M is a matrix representing masks. $M_{ij} = -\infty$ indicates that the j-th token has no contribution to the output of the i-th token, so it is essentially "*masked out*" when encoding the i-th token. Therefore, by setting $M_{ij} = -\infty$ for all $j > i$, we can calculate all outputs simultaneously without looking at future contexts. It was pre-trained on the WebText dataset consisting of 40 GB high quality text crawled from internet sources. We use the small version (12 layers and 117 M parameters) of the released GPT-2 models.

4.3 GPT-2 Fine-Tuning

After the model is pre-trained with a language model objective, it can be fine-tuned on downstream tasks with supervised data. In our task, we adapt the GPT-2 to the target domain by fine-tuning using RCT data. Figure 2 illustrates the learning procedure. By fine-tuning on the target data, the GPT-2 model may have the potential of understanding and generating medical texts.

In the fine-tuning stage, we modify the attention masking of the GPT-2 model so that source byte pairs are fully aware of the entire context of the source sentence, while the target byte pairs are

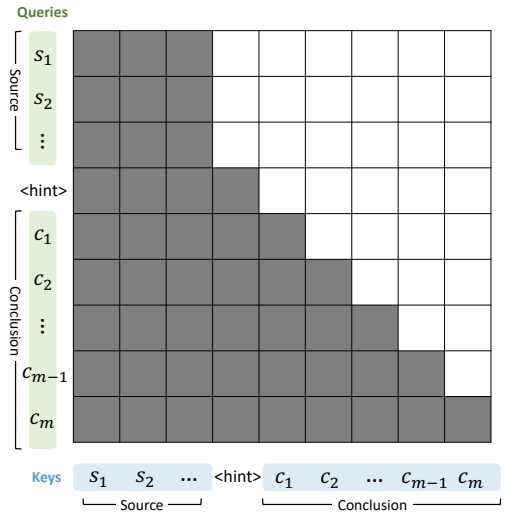


Figure 3: The attention mask used during fine-tuning. White cells denote $-\infty$ elements and grey cells denote 0 in the mask matrix.

aware of the entire source sentence plus the generated byte pairs that precede itself. That is, for context token pairs $(c_i, c_j) \in c_1, \cdots, c_m$, we set $M_{ij} = -\infty$ for all $j > i$, while for context and source token pairs (c_i, s_j), where $c_i \in c_1, \cdots, c_m$ and $s_j \in s_1, \cdots, s_n$, we set $M_{ij} = 0$. For all source token pairs $(s_i, s_j) \in s_1, \cdots, s_n$, we also set $M_{ij} = 0$. This setting is illustrated in Figure 3.

5 Experiments

Here we describe experimental details of the baseline pointer-generator model and the GPT-2 fine-tuned models.

5.1 Experimental Setup

The baseline model is a pointer-generator network (See et al., 2017) with both copy and coverage mechanisms, and is trained with a coverage loss. We adopt the implementation of Zhang et al. (2018). The vocabulary size is about 50,000, with uncased word embeddings pretrained from the PubMed RCT 200k training set and the abstracts from the PubMed dataset of long documents (Cohan et al., 2018). We concatenate $n \in \{0, 1\}$ hint words following the source sentences, where the hint words are first n words of the target conclusion. Our pointer-generator model uses beam search with beam size 5 to decode the final output conclusion.

In our GPT-2 models, we conduct conclusion generation using $n \in \{0, 1, 3, 5\}$ hint words. For $n = 0$, we append "In conclusion , " to the in-

System	ROUGE-1	ROUGE-2	ROUGE-L
PGNet $n = 0$	27.11	7.61	21.87
PGNet $n = 1$	26.88	8.19	22.63
GPT-2 $n = 0$	30.33	11.34	25.14
GPT-2 $n = 1$	**31.61**	**11.88**	**26.71**
GPT-2 $n = 3$	29.94	11.55	25.85
GPT-2 $n = 5$	29.79	11.29	25.94
GPT-2 res	24.24	6.79	20.71

Table 2: ROUGE scores of the PGNet baseline models and the GPT-2 fine-tuned models on the development set. The GPT-2 res were trained with the "results" section only. Addition of $n > 1$ hint words did not show further gains in ROUGE scores.

System	TP	TN	FP	FN	N/A	Acc.
PGnet $n = 1$	15	3	5	3	24	36%
GPT-2 $n = 0$	24	3	4	5	14	54%
GPT-2 $n = 1$	**26**	**6**	**5**	**3**	**10**	**64%**
Target	32	11	0	0	7	86%

Table 3: Human evaluation results for text understanding on the annotation questions of 50 randomly selected source documents. Note that some source documents which don't fit into the binary paradigm of positive or negative results are classified as N/A. TP: True Positive; TN: True Negative; FP: False Positive; FN: False Negative; N/A: Not Applicable.

put. Also, we perform data ablation study using only the "results" section as the model input. To address the memory constraint on our machines, we only train examples that are less than 500 byte pairs after encoding. Because GPT-2 model uses BPE for input and output, the generated conclusions are capitalized. Previous work showed that beam search did not help the generation quality of GPT-2 models (Holtzman et al., 2019), so we simply use greedy decoding to generate the conclusions . Our GPT-2 model is fine-tuned with teacher forcing, using the SGD optimizer with learning rate of 0.001, momentum of 0.9 and the decay factor of 0.0005. Our model is based on a PyTorch implementation of GPT-2 [†].

5.2 Automatic Evaluation

Table 2 shows the best validation ROUGE scores of baselines and our models. Note that the hint words are not considered in score calculation and the output of all models are lower-cased. The GPT-2 fine-tuned model significantly outperforms the pointer-generator (PGNet) baseline on all ROUGE scores, where the best performing model is GPT-2 with hint word $n = 1$, demonstrating the effectiveness of generating good conclusion in our model. However, more hint words do not bring additional gain in ROUGE scores, probably because more constraints hinder the GPT-2 model to explore potentially good conclusions. Moreover, the ablation result shows the significant drop in all ROUGE scores, indicating the importance of including the "background" and "objective" sections in the input for better content understanding.

5.3 Human Evaluation

We recruited 10 medical students with prior training in bio-statistics and epidemiology to annotate and rate the generated conclusions. Our questionnaire contains two types of questions: the annotation question and the rating question.

- **Annotation**: A annotation question contains a source document and four conclusions, namely the target conclusion written by human, the GPT-2 $n = 0$, the GPT-2 $n = 1$ and the PGnet generated conclusions. The raters are asked to classify each generated conclusions as either true positive, true negative, false positive, false negative or not applicable. We define true / false as whether the generated conclusion corresponds to the given document, and positive / negative as whether the intervention studied has a statistically significant effect, regardless of the effect being favourable or detrimental to the patients. This is to explicitly examine whether the generated conclusion is precise in terms of RCT content understanding.

- **Rating**: Rating questions use the same set-up except the question is a 5 point Likert scale for *correctness*, *quality* and *overall* impression. Each rater is given 5 annotation questions and 5 rating questions, with each source document randomly chosen from the validation set. This is to judge the generated conclusions both regarding to and regardless of the source document.

[†] https://github.com/huggingface/pytorch-transformers

System	Correctness	Quality	Overall
PGnet $n = 1$	3.02	2.86	2.86
GPT-2 $n = 0$	**3.42**	3.66	**3.52**
GPT-2 $n = 1$	3.30	**3.94**	3.50
Target	3.92	4.08	3.98

Table 4: Human evaluation results for generation quality on the rating questions.

To mitigate bias, we do not inform which conclusion was generated or written by human, and the conclusions are lower-cased and randomly ordered in each question for fair comparison.

Table 3 presents the results from the annotation question, where the number of true positive and true negative generations from the GPT-2 fine-tuned models increase when compared to the PGNet baseline. It is clear that the proposed GPT-2 achieves better performance in terms of accuracy (the ratio of true samples). We also include the performance of human-written conclusions in the last row, which serves as the upper bound of this task. However, there is still a gap between human-written conclusions and the generated ones.

In the rating questions depicted in Table 4, the human written conclusions obtain a score nearly 4 out of 5 on all three dimensions. The GPT-2 models have comparable scores in overall impression, both scoring around 3.5 out of 5. The most significant improvement of the GPT-2 generated conclusions is the text quality, with the correctness improvement to a lesser extent. The correctness of GPT-2 $n = 1$ is slightly better than that of GPT-2 $n = 0$ in the annotation question, yet in the rating question, GPT-2 $n = 0$ has a higher averaged score. In sum, the human evaluation results demonstrate that our models significantly outperform the baseline pointer generator and tell that the proposed RCT conclusion generation task is not the same as typical summarization task, so deep text understanding is required for better performance.

6 Discussion

From the human evaluation results and our empirical inspection, we discover two major problems concerning the quality of the generated conclusions from GPT-2 models. First, there is some repetition in the generated conclusions, which impair the quality of generated text, though not as common in that of RNN-based models. We suggest additional weighted decoding or coverage mechanisms to avoid such problems. Second, the GPT-2 generated conclusions are significantly shorter than the targets. The average length generated by GPT-2 $n = 0$ and GPT-2 $n = 1$ are 19.4 and 21.0, while that of human written conclusions is 41.4. This could be caused by the limitation of greedy decoding, but the examples generated by PGnet, which applies beam search, only gives an average length of 22.6. This suggests investigation of additional measures to enrich and lengthen the generated conclusions in future work.

Another important issue is the correctness of the generation model. The GPT-2 models are able to identify simple patterns and generate conclusions with the correct relationship. However, errors occur when the study design becomes more complicated or the outcomes are complex. Therefore, future work should aim at enhancing the language understanding capabilities of generation models. Methods such as pre-training the GPT-2 models with medical domain literature or using external background knowledge might fill the missing gap in the correctness performance. This is very crucial regarding to our RCT understanding task and other tasks that require precise and reliable language generation.

Here we select 3 examples to better illustrate our evaluation methods and the discussed limitations of the current models. The example in Table 5 show two successful generations from the GPT-2 models. Table 6 shows a false positive example by the GPT-2 $n = 1$ model. On the other hand, a false negative example generated by the GPT-2 $n = 0$ can be seen in Table 7. The generated conclusions in Table 7 is also much shorter than the target conclusion written by human. Other factors that could cause this issue may include that the human authors mention information not included in the preceding source document, additional comments on the results and background knowledge and they paraphrase the same concept in different ways.

Given the above results, this paper opens a new research direction by formulating the RCT conclusion generation task and investigates the potential of language generation models towards better understanding of medical documents.

Source: Proton pump inhibitor (PPI) therapy is considered as the first choice for treatment of non-erosive reflux disease (NERD) . However , NERD is less sensitive to PPIs than erosive gastroesophageal reflux disease (GERD) and the differences between PPIs and H2 receptor antagonists are less evident in NERD than in erosive GERD . Since gastric acid secretion is lower in the Japanese population than in Western populations , we aimed to investigate whether PPI therapy is really necessary for NERD patients in Japan . Both roxatidine and omeprazole significantly improved the heartburn score at 4 and 8 weeks . The clinical response rates did not differ between roxatidine and omeprazole . Both roxatidine and omeprazole significantly relieved not only reflux but also abdominal pain and indigestion . The degrees of improvement did not differ between the two groups .
Target (True Positive): Roxatidine relieved the symptoms of NERD patients with similar effectiveness to omeprazole . Therefore , roxatidine may be a good choice for NERD treatment .
GPT-2 $n = 0$ **(True Positive):** Both roxatidine and omeprazole significantly improved the heartburn score at 4 and 8 weeks.
GPT-2 $n = 1$ **(True Positive):** Roxatidine and omeprazole are effective in relieving symptoms of NERD in Japanese patients.

Table 5: An example of GPT-2 models generating true positive conclusions.

Source: To evaluate the efficacy of oxcarbazepine (OXC) in the treatment of agitation and aggression in patients with Alzheimer 's disease , vascular dementia or both . In total , 103 institutionalized patients at 35 sites were randomized to the trial . After 8 weeks , no statistically significant differences were found between the 2 groups for all outcomes . A trend was observed in favor of the OXC group in the reduction in the scores on the BARS (p = 0.07) .
Target (True Negative): This study found no significant effect of OXC in treatment of agitation and aggression in patients with dementia .
GPT-2 $n = 0$ **(True Negative):** OXC was not effective in the treatment of agitation and aggression in patients with Alzheimer's disease, vascular dementia or both.
GPT-2 $n = 1$ **(False Positive):** This study suggests that OXC is effective in the treatment of agitation and aggression in patients with Alzheimer's disease.

Table 6: An example of GPT-2 $n = 0$ model generating a true negative conclusion, while the GPT-2 $n = 1$ model generated a false positive one.

Source: Atrial fibrillation (AF) is the most common complication following coronary artery bypass graft (CABG) . The mechanism of AF after CABG is not well defined ; however , it is suggested that endogenous adenosine , released in response to tissue hypoxia , may play a mechanistic role in these arrhythmias . The purpose of this study was to examine whether intravenous theophylline , via adenosine A1 receptor antagonism , would correct or modify new-onset early (<48 h post CABG) atrial fibrillation in patients post CABG , and thereby implicate endogenous adenosine as an inciting agent . Thirty patients comprised the study group . In Group A , 8 of the 15 patients (53 %) converted from AF to sinus rhythm within 15 min of theophylline administration . One patient who converted to sinus rhythm 20 min after theophylline administration was accepted as showing a negative response . In the placebo-treated group , no patient converted to sinus rhythm within 15 min (p <0.007 compared with Group A) .
Target (True Positive): The mechanism of AF after CABG is not well defined and is probably multifactorial . However , this study demonstrated that antagonism of the adenosine A1 receptor can promptly convert many of these patients back to sinus rhythm , and thereby implicates endogenously released adenosine in a mechanistic role for inciting early (<48 h) post-CABG AF .
GPT-2 $n = 0$ **(False Negative):** Intravenous theophylline, via adenosine A1 receptor antagonism, did not improve early AF in patients post CABG.
GPT-2 $n = 1$ **(True Positive):** The results of this study suggest that intravenous theophylline, via adenosine A1 receptor antagonism, may correct or modify early AF in patients post CABG.

Table 7: An example of GPT-2 $n = 0$ model generating a false negative conclusion, while the GPT-2 $n = 1$ model generated a true positive one.

7 Conclusion and Future Work

This work introduces the RCT paper conclusion generation task as a stepping stone to the automatic understanding of clinical research literature. Our results show the general domain pre-trained GPT-2 language model can be fine-tuned to generate medical domain conclusions. The evaluation results show improvements regarding to both quality and correctness in conclusions generated by the

fine-tuned GPT-2 model compared to the pointer-generator summarization model. Further study is needed to enhance the generation quality by reducing repetition errors and increasing the generation length, and to improve the correctness through better language understanding for practical clinical scenarios.

Beyond generating conclusions for RCT papers, generative language models in the medical domain with improved correctness and quality can open up new opportunities to tasks that require profound domain knowledge. For example, automatic generation of systemic review and meta-analysis articles.

Acknowledgements

We would like to thank reviewers for their insightful comments on the paper. This work was financially supported from the Young Scholar Fellowship Program by Ministry of Science and Technology (MOST) in Taiwan, under Grant 108-2636-E002-003 and 108-2634-F-002-015.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *stat*, 1050:21.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Frank Davidoff and Jennifer Miglus. 2011. Delivering Clinical Evidence Where It's Needed: Building an Information System Worthy of the Profession. *JAMA*, 305(18):1906–1907.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Christian Gulden, Melanie Kirchner, Christina Schttler, Marc Hinderer, Marvin Kampf, Hans-Ulrich Prokosch, and Dennis Toddenroth. 2019. Extractive summarization of clinical trial descriptions. *International Journal of Medical Informatics*, 129:114 – 121.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *CoRR*, abs/1904.09751.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141, Melbourne, Australia. Association for Computational Linguistics.

Baoyu Jing, Pengtao Xie, and Eric Xing. 2018. On the automatic generation of medical imaging reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2577–2586, Melbourne, Australia. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and elmo on ten benchmarking datasets. *CoRR*, abs/1906.05474.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Rimma Pivovarov and Nomie Elhadad. 2015. Automated methods for the summarization of electronic health records. *Journal of the American Medical Informatics Association*, 22(5):938–947.

Hemant Pugaliya, Karan Saxena, Shefali Garg, Sheetal Shalini, Prashant Gupta, Eric Nyberg, and Teruko Mitamura. 2019. Pentagon at mediqa 2019: Multitask learning for filtering and re-ranking answers using language inference and question entailment.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 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.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Bonnie Sibbald and Martin Roland. 1998. Understanding controlled trials: Why are randomised controlled trials important? *BMJ*, 316(7126):201.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010. Curran Associates Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Yichong Xu, Xiaodong Liu, Chunyuan Li, Hoifung Poon, and Jianfeng Gao. 2019. Doubletransfer at MEDIQA 2019: Multi-source transfer learning for natural language understanding in the medical domain. *CoRR*, abs/1906.04382.

Yuhao Zhang, Daisy Yi Ding, Tianpei Qian, Christopher D. Manning, and Curtis P. Langlotz. 2018. Learning to summarize radiology findings. In *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, pages 204–213, Brussels, Belgium. Association for Computational Linguistics.