Enhanced Named Entity Recognition in Clinical Trial Abstracts Through Augmented Data Training

Luis Klocke
Deep Learning for Natural Language Processing – SS23 – Philipp Cimiano and Philipp Heinisch

September 29, 2023

1 Introduction

As the rapid digitization of healthcare data continues, it becomes imperative to develop sophisticated Natural Language Processing (NLP) techniques that can effectively interpret and extract valuable information from extensive medical literature. This paper specifically focuses on Named Entity Recognition (NER) in the realm of clinical trial abstracts. In this domain, it is crucial to efficiently extract and categorize information such as dosages, patient allocation and observed results, thereby accelerating the analysis and interpretation of trial results, which is instrumental for advancing personalized treatment strategies and improving patient care.

We establish baseline performances using a traditional Recurrent Neural Network (RNN) model and a more sophisticated BERT-based approach for NER tasks in clinical trial abstracts. Furthermore, we introduce an augmented BERT model which integrates data that has been expanded and refined through augmentation techniques designed to optimize the quality and diversity of the training dataset. This augmentation process utilizes GPT-4, a cutting-edge language model known for its proficiency in generating diverse textual outputs, both for augmenting the existing training abstracts with synonym substitution and value modifications, as well as for the generation of entirely novel abstracts.

Additionally, we incorporated abstracts from the PubMed database that provide relevant context, specifically selecting those pertaining to clinical trials on type 2 diabetes mellitus.

Labeling the augmented abstracts presented considerable challenges. To address this, we developed a unique workflow, utilizing label suggestions derived from original abstracts wherever possible. Additionally, we incorporated predictions generated by a previous iteration of the model for the augmented abstracts. These labels served as preliminary guides, aiding the manual labeling process for each abstract.

To effortlessly generate an abundant amount of training data, we applied random dropout to non-entity tokens within the preceding abstracts, consequently creating slightly varied duplicates of the original texts.

Finally we evaluated the augmented model in comparison to the baseline models using the micro F1 measure and compared the per-label performance of each model.

2 Related work

The application of NER in the field of biomedicine has a long history, tracing back to rule-based approaches and early machine learning techniques (Friedman et al., 1994). Conditional Random Fields introduced a significant advancement in the domain (Settles, 2005) and were further improved with advancements in deep learning. Particularly, the integration of LSTM-CRFs offered a noteworthy improvement to the capabilities of CRFs in NER applications (Habibi et al., 2017).

As the field progressed, the fine-tuning of expansive language models emerged as a pivotal strategy, setting new performance benchmarks for natural language tasks within biomedicine. This shift is epitomized by the advent of BioBERT, which has been instrumental in advancing the state of NER applications in the field (Lee et al., 2019)

In parallel, recent works have explored data augmentation using pre-trained transformer models, as evidenced by recent contributions (Kumar, Choudhary, and Cho, 2021). GPT-4 (OpenAI, 2023) is the current state of the art transformer model and has been shown to be useful for data augmentation, especially in low-ressource settings (Møller et al., 2023).

3 Model

The initial dataset for training consisted of 68 training abstracts (1445 sentences) and 20 test abstracts (429 sentences). Each token within the sentences was annotated with a set of labels selected from a predefined pool of 36 unique labels. Additionally, a special label was designated for padding tokens. The task was formulated as a multi-label classification problem wherein each word was represented by a binary vector. Each element within this vector corresponded to one of the distinct labels, with its value indicating the presence or absence of the respective label assigned to the word.

The baseline Recurrent Neural Network (RNN) was structured with an initial embedding layer of dimensionality 100, succeeded by a singular RNN layer with a hidden dimension of 128, culminating in a dense output layer. To facilitate the model's validation process, 10% of the training data was used as validation data. The training was executed with a learning rate set at 0.001 for a total of 100 epochs.

The baseline BERT model utilized in our study was initialized with pre-trained weights from the "bert-base-uncased" checkpoint. The model was fine-tuned for a span of 20 epochs, employing the AdamW optimizer with an initial learning rate of 0.0001. Furthermore, a learning rate scheduler was implemented to effectuate a linear decay of the learning rate throughout the training epochs.

The augmented BERT model was initialized using the identical checkpoint as the baseline, maintaining consistency in the learning rate, optimizer, and scheduler across both models. The training dataset for the augmented BERT was enriched with additional data: 91 augmented abstracts, 8 generated abstracts, and 25 real abstracts were included alongside the original 68 training abstracts. An additional set of 600 abstracts was generated via the random dropout, applied to non-entity words (dropout rate $p = 0.2$), resulting in a final augmented abstract count of 724, with approximately 408 (56.35%) of those being augmented by GPT-4 (See appendix). We reduced the number of epochs to 8 to account for the large amount of duplicate information now present in the training data. This reduction not only mitigated the risk of overfitting due to repetitive data but also ensured that the model's training runtime remained within practical bounds.

For the two BERT-based models, a deliberate decision was made to forgo the use of a distinct validation set, thus maximizing the size of the available training set. This strategic choice was primarily driven by the augmentation process, which benefited from this approach as it expanded the pool of data available to effectively generate meaningful and diverse training examples. To ensure that the models were adequately trained, we measured the micro f1 test set performance through training (See Figure 1 and 2. Inputs for the BERT-based models were constructed by concatenating sentences from the abstracts, with each input sequence containing as many sentences as could fit within the model's limit on sequence length (512 input ids). This approach was adopted to maximize the contextual information accessible to the models during the training phase, while preserving the inherent structure of individual sentences.

To enhance the training sentences, GPT-4 was employed with a task-specific custom instruction (See appendix for more information). The abstracts generated through this process were subsequently split into individual sentences and tokenized. Predictive labeling of these abstracts was initially performed using a version of the model that had demonstrated superior performance. This approach ensured that improvements in the model would directly contribute to enhancements in the overall workflow.

For the annotation process, a table was constructed that indexed each token alongside its predicted and suggested labels. The suggestions were made using the labels of the original abstract that were used for the augmentation. Annotators could add or remove labels by providing their index span. Their decisions were guided by considering the predicted labels, the suggested alternatives, and their intuitive grasp of the labeling scheme. To streamline this procedure and promote efficiency, annotators were provided with the predicted labels as the default starting point, which they could then adjust as needed. Visual representations of this process can be seen in the appendix.

4 Results

The evaluation of the three models, RNN, BERT-baseline, and BERT-augmented, was carried out over five runs, with average performance scores derived to obtain a robust estimation of the model's capabilities. The primary metric utilized for this comparison is the Micro F1 score.

Micro F1 Scores (Table 1): The RNN model achieved a mean Micro F1 score of 0.5793 (SD = 0.01261, CI = $(0.5602, 0.5962)$), indicating a moderate level of performance with slight variations across the runs.

On the other hand, the BERT-baseline model exhibited superior performance, attaining a mean Micro F1 score of 0.7749 (SD = 0.01404. CI = $(0.7532, 0.7857)$), suggesting consistent and markedly improved results compared to the RNN model.

The BERT-augmented model outperformed both aforementioned models, achieving the highest mean Micro F1 score of 0.8229 with a relatively lower standard deviation of 0.009144. The confidence interval for this model is $(0.8073, 0.8338)$, showcasing its robust and superior performance across the different runs. The BERT-augmented model exhibits a performance boost of approximately 42.05% over the RNN model, and approximately 6.19% compared to the BERT-baseline.

Per-Label Micro F1 Scores (Table 2): The BERT-augmented model consistently demonstrates higher F1 scores across most labels, further validating its superior and robust performance observed in the overall Micro F1 scores. Some labels show significant improvement with the BERT-augmented model, while others have marginal gains, indicating the model's adaptability and effectiveness in handling different types of data. It is crucial to note that the BERT-baseline model never significantly outperformed the BERT-augmented counterpart, emphasizing the positive impact of our data augmentation techniques in enhancing the model's ability to accurately predict various labels. While the BERT-augmented model outperforms the RNN model in most labels by a wide margin, there are some labels that the RNN performs exceptionally well, even outperforming the BERT-based models, such as the labels 'DiffGroupsAbsValue', 'ResultMeasuredValue', 'SdDevResValue' and 'SubGroupDescription'.

5 Conclusion/Discussion

The results of our study demonstrate the effectiveness of the presented data augmentation techniques. These methodologies not only improve the overall performance of the model but also enhance its reliability and accuracy across diverse labels. The BERT-augmented model consistently outperforms the BERT-baseline and RNN models, reflecting a noticeable improvement in Micro F1 scores both overall and on a per-label basis.

Notably, the augmentation techniques contribute to substantial gains for specific labels where traditional models might struggle, underscoring their utility in addressing complex classification challenges.

The empirical results from our experimentation underline the utility of GPT-4-based data augmentation techniques, especially in scenarios with limited data availability. Approximately 56% of our augmented training dataset originated from GPT-4, serving as a testament to the technique's central contribution. This substantial proportion underscores the model's reliance on GPT-4 augmentation to build a more comprehensive and diversified training dataset. While the augmentation demonstrated notable improvements in the model's micro F1 measure across various labels, it is imperative to approach the findings with cautious optimism due to the varying impact of GPT-4 augmentation on different labels. These disparities call for further investigation into optimizing augmentation for each label. Through continuous refinement and objective assessment, GPT-4-based data augmentation may become a vital strategy for enhancing model performance and robustness in limited data environments, offering a practical solution for working with constrained datasets.

Reflecting on the augmentation process, it is pivotal to acknowledge that manual labeling remained a significant bottleneck. While the incorporation of GPT-4 facilitated in generating a substantial

Model	Micro F1
RNN	0.5793
BERT-baseline	0.7749
BERT-augmented	**0.8229**

Table 1: Micro F1 Comparison

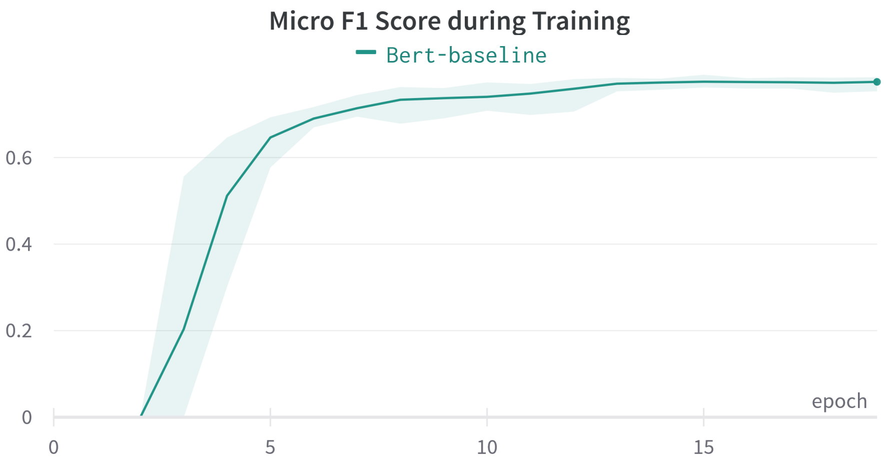


Figure 1: Bert-baseline micro F1 throughout training

volume of augmented data, the need for human intervention to validate and label this generated data not only required considerable time and effort but also underscored the critical need for expertise to ensure the labels' accuracy and relevance. Given these challenges, future iterations of the data augmentation workflow require careful review and improvement. Continuous optimizations in the labeling process will potentially lead to a more efficient and scalable data augmentation workflow, especially in settings with limited data availability.

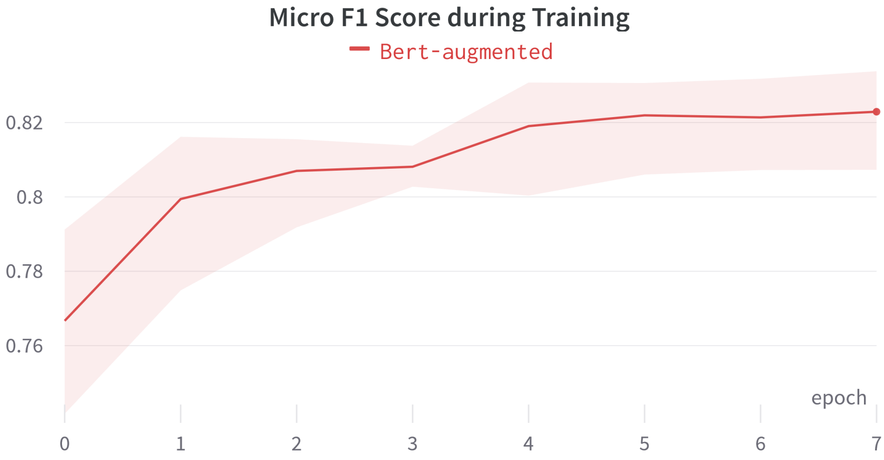


Figure 2: Bert-augmented micro F1 throughout training

Label	**RNN**	**BERT-Baseline**	**BERT-Augmented**
AggregationMethod	0.141	**1**	**1**
AllocationRatio	0.744	**1**	0.977
Author	0	**0.955**	0.936
AvgAge	0.068	0.022	**0.556**
CTDesign	0	0	**0.919**
ConclusionComment	0	0.884	**0.949**
ConfIntervalChangeValue	**0.38**	0.172	0.017
ConfIntervalDiff	0	0.718	**0.784**
Country	**0.005**	0	0
DiffGroupAbsValue	**0.811**	0.578	0.736
DoseDescription	0.085	0.037	**0.172**
DoseValue	0	0.566	**0.69**
Drug	0.635	**1**	**1**
FinalNumPatientsArm	0	**1**	**1**
Frequency	0	0.653	**0.669**
Journal	0.402	**0.99**	0.997
MinAge	0	0	**0.939**
NumberAffected	0.401	0	**0.682**
NumberPatientsArm	0.484	0.768	**0.876**
NumberPatientsCT	0.113	0.722	**0.89**
ObjectiveDescription	0	0.833	**0.886**
ObservedResult	0.034	0.249	**0.254**
PMID	0	**1**	0.997
PValueChangeValue	0.147	0.086	**0.239**
PercentageAffected	0.143	**0.904**	0.895
Precondition	0	0.565	**0.608**
PublicationYear	0	0.99	**1**
PvalueDiff	0	**0.874**	**0.874**
RelativeChangeValue	0.441	**1**	**1**
ResultMeasuredValue	**0.785**	0.478	0.62
SdDevBL	0.069	0	**0.528**
SdDevChangeValue	0.281	0.106	**0.556**
SdDevResValue	**0.668**	0.181	0.324
SubGroupDescription	**0.772**	0	0.199
TimePoint	0.262	0.636	**0.683**
Title	0.588	0.998	**0.999**

Table 2: Average F1 Scores per Label

6 Appendix

Ratio of GPT-4 influenced abstracts in the final training data:

Total Initial Abstracts Before Dropout:

$$68 \text{ (original)} + 91 \text{ (GPT-4 augmented)} + 8 \text{ (GPT-4 generated)} + 25 \text{ (real additional)} = 192$$

Proportion of GPT-4 Abstracts Before Dropout:

$$\frac{99}{192} = 0.5156 \text{ (approximately)}$$

Estimated GPT-4 Influenced Dropout Abstracts:

$$600 \times 0.5156 \approx 309$$

So, approximately 309 dropout abstracts are influenced by GPT-4.

Total Number of GPT-4 Influenced Abstracts:

$$99 \text{ (original GPT-4 generated)} + 309 \text{ (GPT-4 influenced dropout)} = 408$$

Percentage of GPT-4 Influenced Abstracts:

$$\frac{408}{724} \times 100 = 56.35\% \text{ (approximately)}$$

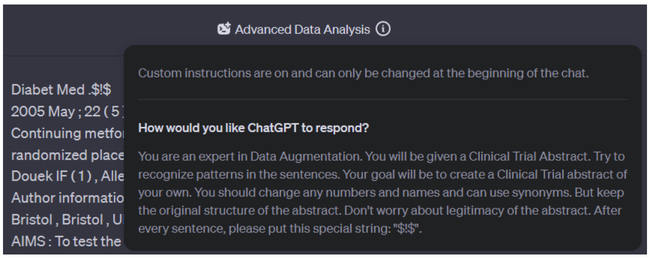
GPT-4 Shared Chat: Abstract Augmentation
`https://chat.openai.com/share/6ffc9d44-de14-4342-b208-8929ab34986c`
Custom Instruction:
You are an expert in Data Augmentation. You will be given a Clinical Trial Abstract. Try to recognize patterns in the sentences. Your goal will be to create a Clinical Trial abstract of your own. You should change any numbers and names and can use synonyms. But keep the original structure of the abstract. Don't worry about legitimacy of the abstract. After every sentence, please put this special string: ”$!$”.

See Figure 3

GPT-4 Shared Chat: Abstract Generation
`https://chat.openai.com/share/5f4501ac-38b9-46b0-aa7a-6f82d856cbf8`
Custom Instruction:
What would you like ChatGPT to know about you to provide better responses?:
I like it if you follow the structure of this example: ”””Diabetes Obes Metab . ! 2014 Nov ; 16 (11) : 1087 - 95 . doi : 10 . 1111 / dom . 12322 . Epub 2014 Jul 8 . $!$ Effect of the sodium glucose co - transporter 2 inhibitor canagliflozin on plasma volume in patients with type 2 diabetes mellitus . $!$ Sha S (1) , Polidori D , Heise T , Natarajan J , Farrell K , Wang SS , Sica D , Rothenberg P , Plum - M ö rschel L . $!$ Author information : (1) Janssen Research & Development , LLC , Raritan , NJ , USA . $!$”””
How would you like ChatGPT to respond?
You are an expert in Data Augmentation. Your goal will be to create a very detailed Clinical Trial abstract about some form of treatment for patients with type 2 diabetes. When writing the abstract, pay attention to keep the structure of the abstract as close as possible to a real abstract as if it had been mined from a medical database. Write in a scientific tone. Content: Include values about dosages, patients allocation, CI, P, sdv (±). Structure: You should start with a journal name and publication information before the title. Here is an example: ”””Diabetes Care . $!$ 2013 May ; 36 (5) : 1304 - 11 . doi : 10 . 2337 / dc12 - 0719 .$!$”””. Don't include keywords. Come up with a similar journal name. You should name the authors and the author information, here is an example: ”””Rosenstock J (1) , Brazg R , Andryuk PJ , Lu K , Stein P ; Sitagliptin Study 019 Group . $!$ Author information : (1) Dallas Diabetes and Endocrine Center , Dallas , Texas , USA .$!$”””. Following the conclusion, you should include the doi, PMCID,

PMID like in this example: """DOI : 10 . 2337 / dc12 - 0719 PMCID : PMC3631843 PMID : 23230096 [Indexed for MEDLINE]$!$""" in a single line. You should not add any additional structuring yourself, like highlighting or separating. The entire abstract including the additional information should be just a text. After every single sentence and newline that you output, including the information about the abstract, please put this special string: "$!$"

See Figure 4 and 5

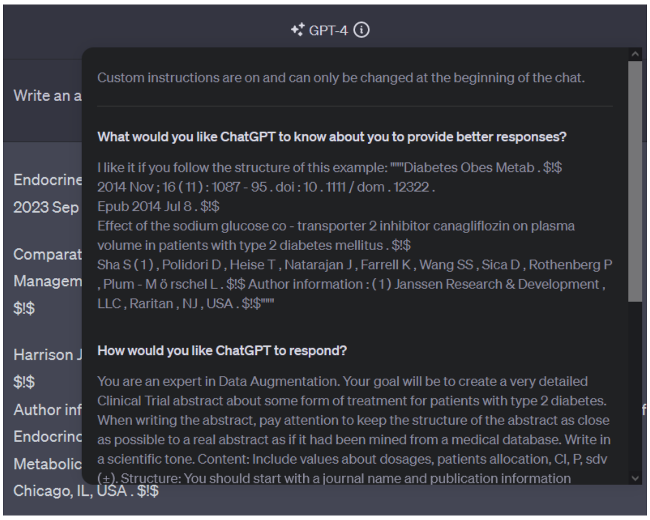


Custom instructions are on and can only be changed at the beginning of the chat.

How would you like ChatGPT to respond?

You are an expert in Data Augmentation. You will be given a Clinical Trial Abstract. Try to recognize patterns in the sentences. Your goal will be to create a Clinical Trial abstract of your own. You should change any numbers and names and can use synonyms. But keep the original structure of the abstract. Don't worry about legitimacy of the abstract. After every sentence, please put this special string: "$!$".

Figure 3: Abstract Augmentation Custom Instruction



Custom instructions are on and can only be changed at the beginning of the chat.

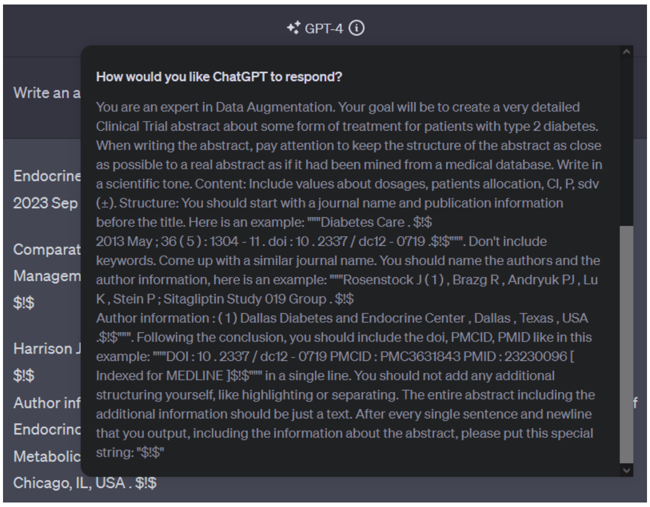
What would you like ChatGPT to know about you to provide better responses?

I like it if you follow the structure of this example: """Diabetes Obes Metab . $!$
2014 Nov ; 16 (11) : 1087 - 95 . doi : 10 . 1111 / dom . 12322 .
Epub 2014 Jul 8 . $!$
Effect of the sodium glucose co - transporter 2 inhibitor canagliflozin on plasma volume in patients with type 2 diabetes mellitus . $!$
Sha S (1) , Polidori D , Heise T , Natarajan J , Farrell K , Wang SS , Sica D , Rothenberg P , Plum - M ö rschel L . $!$ Author information : (1) Janssen Research & Development , LLC , Raritan , NJ , USA . $!$"""

How would you like ChatGPT to respond?

You are an expert in Data Augmentation. Your goal will be to create a very detailed Clinical Trial abstract about some form of treatment for patients with type 2 diabetes. When writing the abstract, pay attention to keep the structure of the abstract as close as possible to a real abstract as if it had been mined from a medical database. Write in a scientific tone. Content: Include values about dosages, patients allocation, CI, P, sdv (±). Structure: You should start with a journal name and publication information

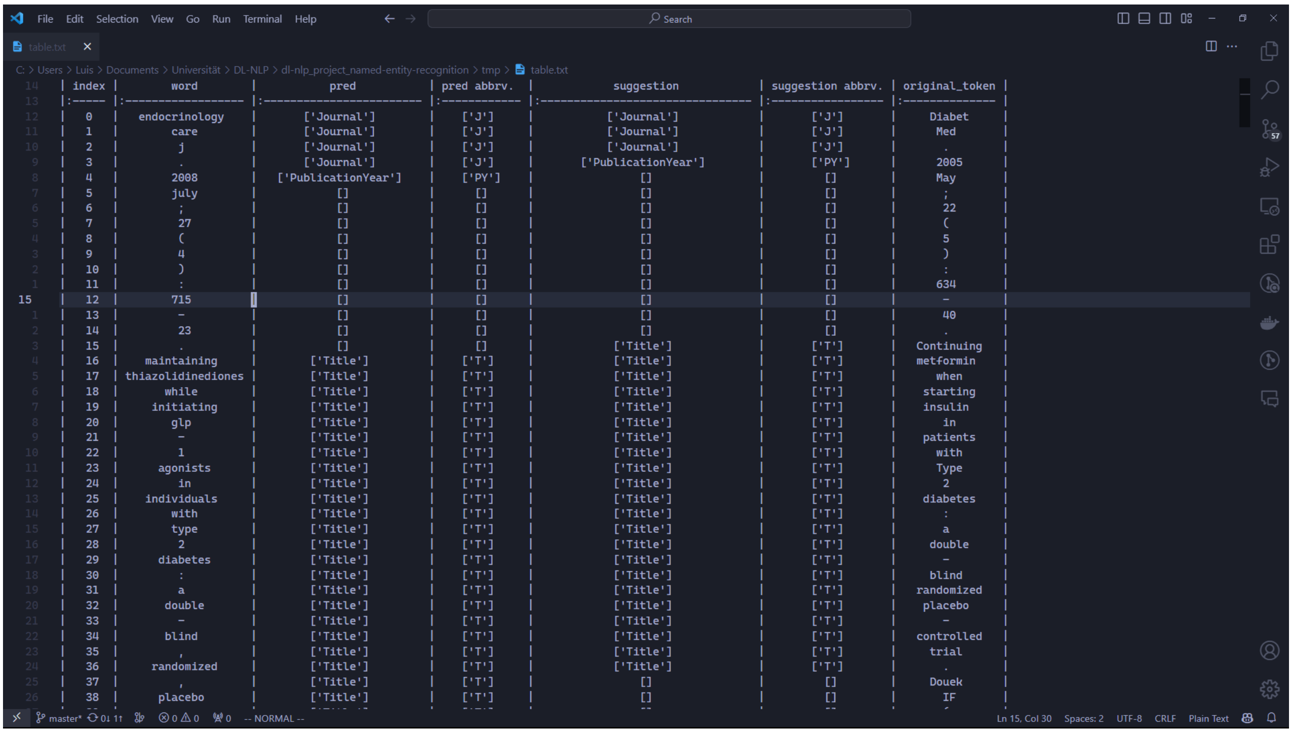
Figure 4: Abstract Generation Custom Instruction 1



How would you like ChatGPT to respond?

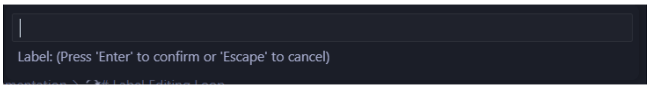
You are an expert in Data Augmentation. Your goal will be to create a very detailed Clinical Trial abstract about some form of treatment for patients with type 2 diabetes. When writing the abstract, pay attention to keep the structure of the abstract as close as possible to a real abstract as if it had been mined from a medical database. Write in a scientific tone. Content: Include values about dosages, patients allocation, CI, P, sdv (±). Structure: You should start with a journal name and publication information before the title. Here is an example: """Diabetes Care . $!$
2013 May ; 36 (5) : 1304 - 11 . doi : 10 . 2337 / dc12 - 0719 .$!$""". Don't include keywords. Come up with a similar journal name. You should name the authors and the author information, here is an example: """Rosenstock J (1) , Brazg R , Andryuk PJ , Lu K , Stein P ; Sitagliptin Study 019 Group . $!$
Author information : (1) Dallas Diabetes and Endocrine Center , Dallas , Texas , USA .$!$""". Following the conclusion, you should include the doi, PMCID, PMID like in this example: """DOI : 10 . 2337 / dc12 - 0719 PMCID : PMC3631843 PMID : 23230096 [Indexed for MEDLINE]$!$""" in a single line. You should not add any additional structuring yourself, like highlighting or separating. The entire abstract including the additional information should be just a text. After every single sentence and newline that you output, including the information about the abstract, please put this special string: "$!$"

Figure 5: Abstract Generation Custom Instruction 2



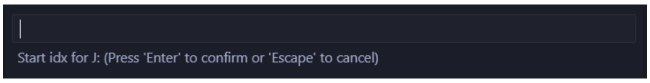
index	word	pred	pred abbrv.	suggestion	suggestion abbrv.	original_token
0	endocrinology	['Journal']	['J']	['Journal']	['J']	Diabet
1	care	['Journal']	['J']	['Journal']	['J']	Med
2	j	['Journal']	['J']	['Journal']	['J']	.
3	.	['Journal']	['J']	['PublicationYear']	['PY']	2005
4	2008	['PublicationYear']	['PY']	[]	[]	May
5	july	[]	[]	[]	[]	;
6	;	[]	[]	[]	[]	22
7	27	[]	[]	[]	[]	(
8	(	[]	[]	[]	[]	5
9	4	[]	[]	[]	[]	)
10	)	[]	[]	[]	[]	:
11	:	[]	[]	[]	[]	634
12	715	[]	[]	[]	[]	–
13	–	[]	[]	[]	[]	40
14	23	[]	[]	[]	[]	.
15	.	[]	[]	['Title']	['T']	Continuing
16	maintaining	['Title']	['T']	['Title']	['T']	metformin
17	thiazolidinediones	['Title']	['T']	['Title']	['T']	when
18	while	['Title']	['T']	['Title']	['T']	starting
19	initiating	['Title']	['T']	['Title']	['T']	insulin
20	glp	['Title']	['T']	['Title']	['T']	in
21	–	['Title']	['T']	['Title']	['T']	patients
22	1	['Title']	['T']	['Title']	['T']	with
23	agonists	['Title']	['T']	['Title']	['T']	Type
24	in	['Title']	['T']	['Title']	['T']	2
25	individuals	['Title']	['T']	['Title']	['T']	diabetes
26	with	['Title']	['T']	['Title']	['T']	:
27	type	['Title']	['T']	['Title']	['T']	a
28	2	['Title']	['T']	['Title']	['T']	double
29	diabetes	['Title']	['T']	['Title']	['T']	–
30	:	['Title']	['T']	['Title']	['T']	blind
31	a	['Title']	['T']	['Title']	['T']	randomized
32	double	['Title']	['T']	['Title']	['T']	placebo
33	–	['Title']	['T']	['Title']	['T']	–
34	blind	['Title']	['T']	['Title']	['T']	controlled
35	,	['Title']	['T']	['Title']	['T']	trial
36	randomized	['Title']	['T']	['Title']	['T']	.
37	,	['Title']	['T']	[]	[]	Douek
38	placebo	['Title']	['T']	[]	[]	IF

Figure 6: Labeling Table



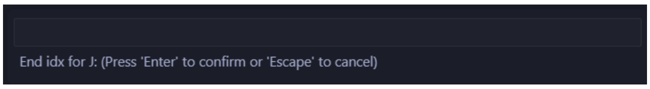
Label: (Press 'Enter' to confirm or 'Escape' to cancel)

Figure 7: Label Input



Start idx for J: (Press 'Enter' to confirm or 'Escape' to cancel)

Figure 8: Label Start Index Input



End idx for J: (Press 'Enter' to confirm or 'Escape' to cancel)

Figure 9: Label End Index Input

References

Friedman, Carol et al. (Mar. 1994). “A general natural-language text processor for clinical radiology”. In: *Journal of the American Medical Informatics Association* 1.2, pp. 161–174. DOI: 10.1136/jamia.1994.95236146.

Habibi, Maryam et al. (July 2017). “Deep learning with word embeddings improves biomedical named entity recognition”. In: *Bioinformatics* 33.14, pp. i37–i48. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btx228. eprint: https://academic.oup.com/bioinformatics/article-pdf/33/14/i37/50314882/bioinformatics_33_14_i37.pdf. URL: https://doi.org/10.1093/bioinformatics/btx228.

Kumar, Varun, Ashutosh Choudhary, and Eunah Cho (2021). *Data Augmentation using Pre-trained Transformer Models*. arXiv: 2003.02245 [cs.CL].

Lee, Jinhyuk et al. (Sept. 2019). “BioBERT: a pre-trained biomedical language representation model for biomedical text mining”. In: *Bioinformatics* 36.4. Ed. by Jonathan Wren, pp. 1234–1240. DOI: 10.1093/bioinformatics/btz682. URL: https://doi.org/10.1093%2Fbioinformatics%2Fbtz682.

Møller, Anders Giovanni et al. (2023). *Is a prompt and a few samples all you need? Using GPT-4 for data augmentation in low-resource classification tasks*. arXiv: 2304.13861 [cs.CL].

OpenAI (2023). *GPT-4 Technical Report*. arXiv: 2303.08774 [cs.CL].

Settles, Burr (Apr. 2005). “ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text”. In: *Bioinformatics* 21.14, pp. 3191–3192. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bti475. eprint: https://academic.oup.com/bioinformatics/article-pdf/21/14/3191/48971532/bioinformatics_21_14_3191.pdf. URL: https://doi.org/10.1093/bioinformatics/bti475.