Comparing Generic and Expert Models for Genre-Specific Text Simplification

Zihao LI[1] and **Matthew Shardlow**[1] and **Fernando Alva-Manchego**[2]
[1] Manchester Metropolitan University
[2] School of Computer Science and Informatics, Cardiff University, UK
21443696@stu.mmu.ac.uk, m.shardlow@mmu.ac.uk,
alvamanchegof@cardiff.ac.uk

Abstract

We investigate how text genre influences the performance of models for controlled text simplification. Regarding datasets from Wikipedia and PubMed as two different genres, we compare the performance of genre-specific models trained by transfer learning and prompt-only GPT-like large language models. Our experiments showed that: (1) the performance loss of genre-specific models on general tasks can be limited to 2%, (2) transfer learning can improve performance on genre-specific datasets up to 10% in SARI score from the base model without transfer learning, (3) simplifications generated by the smaller but more customized models show similar performance in simplicity and a better meaning preservation capability to the larger generic models in both automatic and human evaluations.

1 Introduction

Controllable text simplification is a technique whereby the features of a generated simplification (e.g. its length) can be determined at inference time. Control tokens prepended to the input with specific features' values can be regarded as a way of prompting text simplification systems to generate outputs with certain desired characteristics. This gives rise to flexible and controllable simplification systems that satisfy various demands from different user groups or scenarios with regulated output (Kikuchi et al., 2016; Scarton and Specia, 2018; Nishihara et al., 2019; Martin et al., 2019; Maddela et al., 2021). A use case for such types of models is making specialised information (e.g. related to medicine) more accessible to lay users.

We present genre-specific text simplification research alongside a study on the effects of different genres. We followed the idea of Multilingual Unsupervised Sentence Simplification (MUSS) (Martin et al., 2020), which is the State-of-the-art (SOTA) of controlled text simplification, to build the base model and expert models. Different from MUSS, in which the authors combined the explicit control tokens with the mined paraphrase corpus, we combined the control tokens with two small expert-level genre-specific training subsets derived from Simple TICO-19 corpus (Shardlow and Alva-Manchego, 2022). The base model reimplements the MUSS without the fine-tuning on the paraphrase corpus, while the expert models are further fined-tuned on the genre-specific training subsets.

We choose the newly published Simple TICO-19 dataset (Shardlow and Alva-Manchego, 2022) as our training and test bench of genre-specific tasks for the expert models, because of the manual simplification from experienced annotators and expert-level information in COVID-19. Based on Simple TICO-19, we created the two subsets with unified data source labels as two different genre-specific corpora and designed the genre-specific tasks with different permutations of each kind of subset.

To verify the improvement before and after transfer learning, we tested the performance of the expert models over the base model in the above-mentioned genre-specific scenarios. In addition, considering the strong competitiveness of more updated and larger language models than the base model (Radford et al., 2018; Devlin et al., 2019; Liu et al., 2019; Raffel et al., 2019; Brown et al., 2020), it is worth finding whether the large language models targeting generic content can outperform lightweight custom models that have been specialized for specific tasks. Thus, we also compared the expert models with the leading generic models for generative NLP, covering GPT-3 (Brown et al., 2020) and ChatGPT.

In this paper, we leveraged a newly published text simplification dataset, designed a test scenario for controlled text simplification with different genres, proved the effects of transfer learning on the genre-specific datasets, compared the performance of generic and expert models in SARI score and

BERTScore, and discussed the cost-effectiveness between expert models and generic models.

2 Related Work

Text simplification consists of reducing linguistic complexity at both syntactic and lexical levels without significant loss in the main content (Alva-Manchego et al., 2020b). In practice, this task can be treated as monolingual machine translation (Zhu et al., 2010; Wubben et al., 2012). Research in English highly relies on Wikipedia and Simple Wikipedia (Zhu et al., 2010; Coster and Kauchak, 2011; Woodsend and Lapata, 2011; Kauchak, 2013; Zhang and Lapata, 2017). High-quality manually-made corpora are rare and some may come with restrictions (Xu et al., 2015; Alva-Manchego et al., 2020a; Shardlow and Alva-Manchego, 2022). To alleviate this problem, we combined the large automated corpus with the small manual-made corpus.

Text simplification researchers have recently turned to larger pre-trained language models (Peters et al., 2018; Martin et al., 2020; Omelianchuk et al., 2021; Lu et al., 2021; Sheang and Saggion, 2021; Štajner et al., 2022). From Long-short term memory (LSTM) to transformer-based pre-trained models (Hochreiter and Schmidhuber, 1997; Raffel et al., 2019; Lewis et al., 2020), the order of magnitude of parameters used in models for text simplification has increased dramatically, The parameter count in Bidirectional Auto-Regressive Transformers (BART) is 140 million (Lewis et al., 2020), the value in Text-to-Text Transfer Transformer (T5) reaches 220 million (Raffel et al., 2019), the value in GPT-3 increases to 170 billion (Brown et al., 2020), and the value in Switch Transformer even reaches an astonishing 1.6 trillion (Fedus et al., 2021). With the advent of pre-trained language models in NLP, the SOTA of many common tasks and leaderboard is refreshed (Schwartz et al., 2014; Rajpurkar et al., 2016; Wang et al., 2018). Models with more parameters tend to perform better on downstream tasks (Kaplan et al., 2020). However, larger models require more energy to run (Puvis de Chavannes et al., 2021) and are inaccessible to a typical researcher, hampering reproducibility. Besides, the correlation between large models and high performance is still worth exploring and the necessity of extremely huge models is questionable. To find out the exact situation in text simplification, we leveraged the latest pre-trained large language model ChatGPT.

In addition to the general models, there are also researches focusing on controlled text simplification (Martin et al., 2019, 2020; Sheang and Saggion, 2021). Due to the various demands of lay users in text simplification, the generic output can hardly satisfy the main user group (Xu et al., 2015). Controlled text simplification is introduced to satisfy the various demands of different user groups or in different scenarios with explicit or implicit restraints on the output. In AudienCe-CEntric Sentence Simplification (ACCESS), Martin et al. (Martin et al. (2019)) present the 4 control tokens used in this paper, Sheang and Saggion (Sheang and Saggion (2021) replace the BART model (Lewis et al., 2020) with T5 model(Raffel et al., 2019), further extend the control tokens to 5 and refresh the SOTA. The performance and flexibility of controlled text simplification make it possible to compete with the large pre-trained language models, and they will be tested in this paper.

3 Methodology

In this section we describe the experiments that were undertaken. A visual representation of our methodology is provided in Figure 1, which is explained in further detail throughout the following subsections.

3.1 Datasets

Wikilarge. The Wikilarge dataset (Zhang and Lapata, 2017) is one of the biggest parallel complex-simple sentence datasets based on various existing corpora and contains 296,402 sentence pairs in the training set. We use this training set to fine-tune the base models in this paper.

Simple TICO-19. We leveraged a newly published dataset, simple TICO-19 (Shardlow and Alva-Manchego, 2022) as the test bench for genre-specific simplification, which is based on the dataset: Translation Initiative for COVID-19 (TICO-19) (Anastasopoulos et al., 2020). This dataset contains translations and simplifications related to COVID-19 from multiple resources. Simple TICO-19 contains 3,173 parallel sentences in both English and Spanish. Only the English section is applied in this paper. We split this dataset based on the data source and regard the subsets from different sources as different genres. The subsets are further divided into training, validation and test sets for the expert models.

ASSET. The Abstractive Sentence Simplification Evaluation and Tuning dataset (ASSET) (Alva-Manchego et al., 2020a) is widely used to evaluate the performance of text simplification models. The dataset contains validation and test sets, both are equipped with 10 reference sets. Only the test sets are used as a general test benchmark for both generic and expert models.

3.2 Metrics and Evaluation

We use **SARI score** (Xu et al., 2016) as the main metric for evaluating the simplicity of our systems outputs. It compares the output with reference sentences and calculates the F1-score of *add*, *keep* and *delete* operations from system output compared to the reference sentences. Although there have been criticisms of the metric (Alva-Manchego et al., 2021) recently, it is still the most widely used automatic metric in the evaluation of text simplification (Alva-Manchego et al., 2020b). To increase the reliability of our results, we also include other automatic metrics and human evaluation.

It is worth noting that there is only one reference sentence per instance in Simple TICO-19 and its subsets for genre-specific tasks. This differs from other datasets with multiple references such as ASSET. Thus, the SARI score of uncomparable among different test sets, and the reliability of SARI for Simple TICO-19 may be lower compared with ASSET.

BERTScore (Zhang et al., 2019) is a metric that measures the likelihood between the output and reference sentences. It is calculated by maximizing the cosine distance in vector spaces in the most possible likelihood matrix. According to Scialom et al. (2021), BERTScore has a higher correlation to human evaluation than SARI and shows how similar the output and references are in the aspect of meaning instead of words. We apply BERTScore as a co-reference in both general and genre-specific tasks.

Human evaluation. We also conduct a human evaluation for the results of the genre-specific experiments as the gold reference, compared to the automatic evaluation metrics. We recruited 17 human annotators via Amazon Mechanical Turk. The annotators were selected to have the 'Masters' qualification, indicating that they are trusted workers on the platform. All annotators reported an educational level of undergraduate or above. Twelve annotators are non-native English speakers, whereas five are native speakers of English. Each annotator was presented with 20 instances. Each instance contained an original sentence and a pair of corresponding simplifications from either the generic or expert models, whose order is random to avoid bias. Annotators were asked to evaluate the following two questions on a 5-point Likert scale:

> 1) *Simplicity*: To what extent do you agree the simplified sentence is easy to understand?
>
> 2) *Meaning preservation*: To what extent do you agree the simplified sentence keeps the important information?

There is a total of 340 instances with 50% overlap in the adjacent forms to ensure a more comprehensive score from two annotators. For disagreement, we use the average value as the final score. The results are shown in Table 6 and the sample form is shown in Appendix A.

3.3 Preprocessing

Following the MUSS implementation (Martin et al., 2020), the four control tokens are introduced as follow:

- ⟨DEPENDENCYTREEDEPTH_x⟩ **(DTD)** representing syntactic complexity
- ⟨WORDRANK_x⟩ **(WR)** representing lexical complexity
- ⟨REPLACEONLYLEVENSHTEIN_x⟩ **(LV)** representing the token difference ratio
- ⟨LENGTHRATIO_x⟩ **(LR)** representing the difference in length

Each control token is calculated by comparing the above ratios in complex-simple sentence pairs. After the calculation of the control tokens for the training set, the calculated value of complex sentences is added as a prompt to the beginning of the corresponding complex sentences. The value of these control tokens is rounded to 0.05 and limited in the range of 0.2 to 1.5, except for the LV, which is limited from 0.2 to 1.

In Simple Tico-19 (Shardlow and Alva-Manchego, 2022), due to the manual translation, there are some sentences marked as sentences that require no more simplification. These were removed in the following experiments. The number

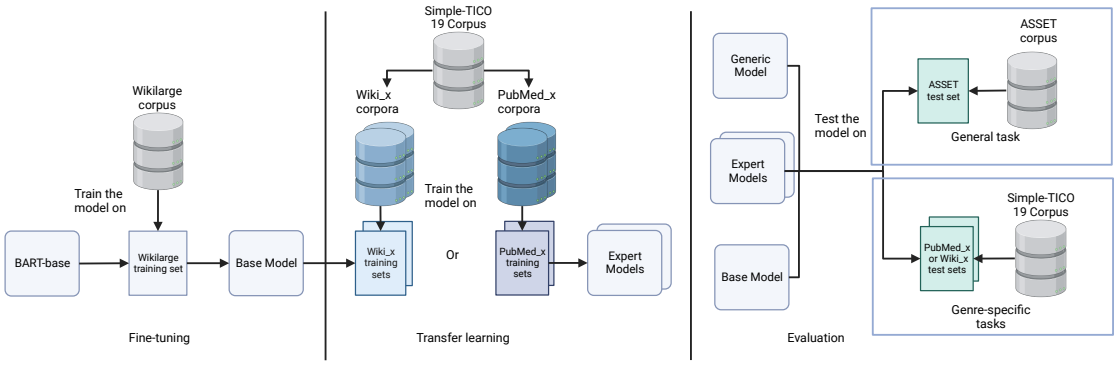

Figure 1: The methodology is represented in three sections. In the left section, we fine-tune BART-base on the WikiLarge training set to give the **base model**. In the middle section, we regard the task as transfer learning and further fine-tune the base model on our Wikipedia_x and PubMed_x training sets to generate the **expert model(s)**. In the right section, we add 2 zero-shot **generic models** through publicly available APIs. We then evaluate our base model, expert models and generic models in the generic simplification task (The Asset test set) and the genre-specific tasks (the Wikipedia_x and Pubmed_x test sets) and compare the results for the models.

Data source	Number of instances
CMU	122
PubMed	809
Wikinews	76
Wikivoyage	206
Wikipedia	1224
Wikisource	101

Table 1: Number of instances in each data source

of instances after the filtering of each data source is shown in Table 1.

Considering the target audience and sentence count, we choose the **PubMed** and **Wikipedia** subsets as representative of two different genres, namely public literature and academic literature, to be applied in the genre-specific tasks. To create training, validation and test sets, we further randomly split the **PubMed** and **Wikipedia** subsets into 3 sections in a ratio of 8:1:1 with a random seed. The generated permutations of the two subsets with a certain random seed $\boldsymbol{x}$ is then marked as **PubMed*x*** and **Wikipedia*x***, such as **PubMed0** and **Wikipedia0**. As a result, there are 978, 122 and 124 sentence pairs in each **Wikipedia*x*** permutation and 647, 81 and 81 sentence pairs in each **PubMed*x*** permutation as train, validation and test set respectively.

3.4 Models for Text Simplification

In this paper, we propose to compare the performance among three versions of a text simplification model: the base model, the generic model and the expert model.

The **base model** is based on BART-base (Lewis et al., 2020) with 6 layers in both encoder and decoder and 140 million parameters. The base model is fine-tuned on the training set of Wikilarge (Zhang and Lapata, 2017) only with the above-mentioned 4 control tokens. The following hyper-parameters were applied: Learning rate: 2e-5, Weight Decay: 0.01, Training epochs: 10. After fine-tuning, the training loss reaches 0.85 without overfitting. By comparing the SARI score of our model on the ASSET test set (Alva-Manchego et al., 2020a) with the original results in MUSS (Martin et al., 2020), it is reasonable to claim that it has reached to the designed performance level.

For **generic models**, we apply the GPT-3 (Brown et al., 2020) and ChatGPT via the API and online platform by OpenAI. Instead of training or fine-tuning, we leverage the 2 models as zero-shot models by promoting. The prompt is set to "Please simplify this sentence for me: " and will be added to the beginning of each complex sentence, then the model will try to generate a simplified version of the input text after the colon. The exact model prompted in the GPT-3 is called "text-davinci-003", which is the latest version, the parameters are set as follows: temperature: 1, frequency_penalty: 0, presence_penalty: 0.

As for ChatGPT, due to the fast iteration speed, the only information available is "ChatGPT Jan 9

Version". During our experiment, since there is no official API released, we accessed the ChatGPT via a fake web browser with session IDs to request responses in batches. The ChatGPT is then accessed on the online platform in the conversations automatically. There is no guarantee of performance compared to the results of API access and different versions of ChatGPT.

The **expert model(s)** are composed of base models after transfer learning on corresponding permutations of subsets. By fine-tuning the pre-trained model on the preprocessed Wikilarge training set (Zhang and Lapata, 2017), the base model learns how to generate simplifications based on the value of control tokens. To leverage the base model as an expert text simplification model, we further fine-tune the model on the preprocessed training set of **Wikipedia*x*** and **PubMed*x*** and then have the corresponding expert models for each permutation of **Wikipedia*x*** and **PubMed*x***. The setting of fine-tuning hyper-parameters is the same as fine-tuning the base model. In the experiment, we build 30 expert models from different permutations of **Wikipedia*x*** and 30 from **PubMed*x***. Due to time constraints, we only evaluate the performance of expert models over 20 permutations of subsets for each genre. In total, we have 40 permutations of subsets with 31 expert models evaluated on each dataset permutation.

3.5 Optimization

Since the values of control tokens influence the quality of the generated output and overall model performance, it is necessary to find an optimal value of the control tokens for the model on the test sets. This is in line with the previous state of the art, but does mean that the results reported are specific to the given test set and alternative parameters may be optimal for another dataset. The value options of most control tokens fall between 0.2 to 1.5 (or 0 to 1 for Levenshtein), so there is only finite options are provided during optimization, and the optimization problem is reduced to finding the best value combination of control tokens within the optimization budget. The optimization budget limits the total number of attempts to find the set of values of control tokens to maximize the metric, which is set to the SARI score. The optimization budget for the general tasks on the ASSET valid set (Alva-Manchego et al., 2020a) is 128, while the value for genre-specific tasks on the valid sets of permutations of **Wikipedia*x*** and **PubMed*x*** is reduced to 64 for time-saving. We used Nevergrad (Rapin and Teytaud, 2018) to find out the local optimal value within the budgets.

3.6 Genre-specific Experiments

To verify the effect of transfer learning, we computed the SARI score on the test set of **PubMed*x***s and **Wikipedia*x***s. Since there is only one reference sentence in the Simple TICO-19, the SARI score on these test sets is only applicable and comparable within the experiment. We tested the base model, generic model and expert models on the test sets from 20 permutations of **PubMed*x***s and **Wikipedia*x***s. For expert models from the same genre of the test set, we only evaluate the expert model trained on the corresponding training set of the test set to avoid data leakage. The average of these models is reported as 'Average corresponding ⟨genre⟩ models' in Tables 3 and 4. As for the expert models from the other genre, we tested 30 expert models from different permutations. The overall results are shown in Table 3 and 4, and the details are shown in Figures 2 and 3. The full results are available in Appendix B.

4 Results

4.1 General task

	Model	SARI	BERTScore
Base	BART-base	44.05	0.777
Generic	GPT-3	41.73	0.703
	ChatGPT	**46.42**	0.731
Expert	**Wikipedia0**	43.24	**0.835**
	PubMed0	43.67	0.812

Table 2: SARI and BERTScore on ASSET test

Table 2 shows the SARI scores and BERTScores on the ASSET test set. ChatGPT reaches the highest SARI score known so far on the ASSET test set, while the expert model **Wikipedia0** obtains the highest BERTScore. Compared to the base model, GPT-3 attains a lower SARI score, whereas ChatGPT attains an improved SARI score. However, the BERTScore is lower for both generic models compared to the base model. Within the 2 general models, the ChatGPT outperforms the GPT-3 in both metrics, which aligns with the model structure and scale. As for the expert models, we find that the SARI scores on the general task drop marginally,

	Model	SARI	BERTScore
Base	BART-base	40.78	0.741
Generic	GPT-3	29.03	0.530
	ChatGPT	31.12	0.542
Expert	Average corresponding expert **Wikipedia** models	**44.30**	**0.756**
	Average **PubMed** models	42.75	0.741

Table 3: Average SARI and BERTScore on all **Wikipedia*x***

	Model	SARI	BERTScore
Base	BART-base	40.56	0.723
Generic	GPT-3	30.72	0.547
	ChatGPT	31.55	0.515
Expert	Average Corresponding expert **PubMed** models	**45.05**	**0.741**
	Average **Wikipedia** models	43.38	0.726

Table 4: Average SARI and BERTScore on all **PubMed*x***

while **Wikipedia0** shows the highest BERTScore among all models.

4.2 Genre-specific task

Table 3 shows the average SARI and BERTScores over all 20 test sets of different permutations from different models. The first row shows the average SARI and BERTScore of the base model, which is only fine-tuned on the WikiLarge training set. The following two rows show the SARI and BERTScore of two generic models on the test sets. The last two rows show the SARI and BERTScore of all expert models. The corresponding **Wikipedia** or **Pubmed** models refer to the corresponding expert models after transfer learning on the training sets (e.g., model **Wikipedia0** to test set **Wikipedia0** and model **Pubmed19** to test set **Pubmed19**). The last row shows a combined average SARI and BERTScore of expert models trained in the other genre. The detailed SARI and BERTScore can be found in Appendix B. The same rules also apply to Table 4.

In both Table 3 and 4, the corresponding expert models, which is the expert model transfer learned on the corresponding training set, have the highest SARI and BERTScore. Although the generic models show very competitive performance in the general task, the lack of fine-tuning led to lower performance in terms of SARI score in the genre-specific scenarios. The fine-tuned models also take advantage of learning the text style in the training set. The overall performance gap between the two generic models is aligned to the gap in Table 2. As for the expert models, they have a much higher SARI score and appear to have a much higher performance, but the actual performance gap between the generic models and expert models needs further exploration. What the SARI score can tell is how they benefit from the transfer learning compared to the base model. It is surprising to see the improvement for both kinds of expert models, which is presumably caused by the sharing characteristics in the two subsets (both are related to Covid-19 information). As a result, the improvement of the overall SARI score for expert models shows the effectiveness of transfer learning for genre-adaptive text simplification.

We also evaluated BERT-score for our generic and expert models on the expert datasets. The BERTScore similarly shows that the simplifications produced by generic models in the expert setting are of worse quality than those produced by the expert models. In Table 3, we notice that there is an improvement in BERTScore on the corresponding expert models over the base model, while no improvement on the average **PubMedx** models in the other genre. The base model was also fine-tuned on the Wikilarge, which belongs to the same genre of the **Wikipediax** models. This may explain why there was no performance gain for the **PubMedx** models. In Table 4, both kinds of expert models gain improvement when measured against the Base model. The genre-specific PubMed expert models attain a higher BERTScore than those fine-tuned on the Wikipedia subsets.

4.3 Detailed SARI score in genre-specific task

Generally, the detailed SARI score is aligned with the overall performance. The corresponding expert model outperforms the other four models in the SARI score across all permutations and the generic models have a much lower SARI score than the base model. The SARI score also shows some similarities among models. We listed the detailed SARI score in Figures 2 and 3 and the remaining tables.

Figure 2 shows the SARI score of 20 **Wikipediax** test sets. Most models follow the order of average score, except for text sets **Wikipedia0**,

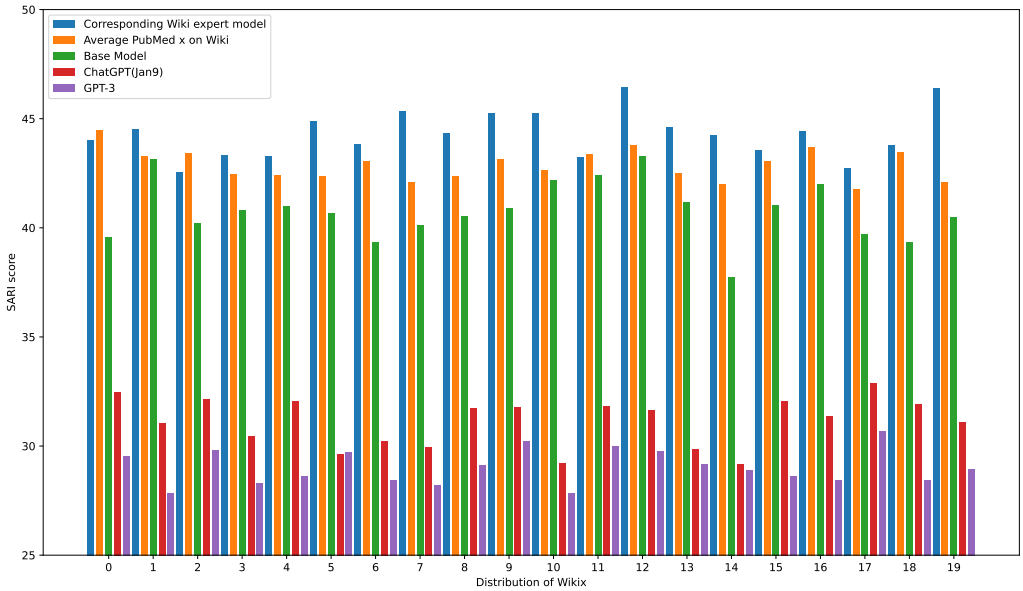

Figure 2: SARI score on **Wikipedia*x*** for expert models

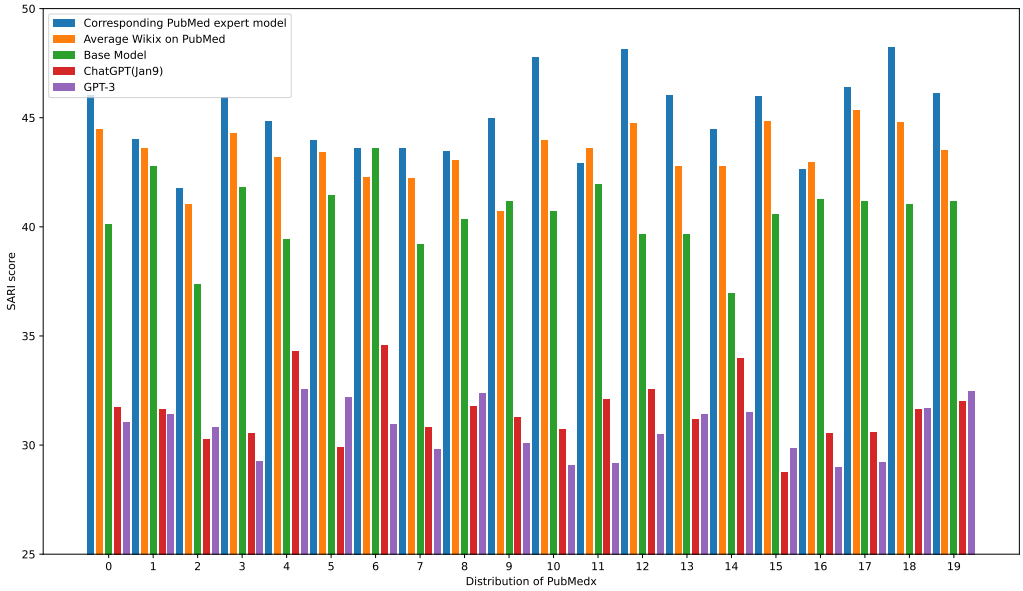

Figure 3: The SARI score on **PubMed*x*** for expert models

Wikipedia2 and **Wikipedia11**. In the above-mentioned test sets, the average SARI score of the other genre outperforms the corresponding expert model. This may be caused by the similarity between the training sets in the other genre and test sets. The fluctuation of the SARI score demonstrates the effect of permutation in the genre-specific experiments and also shows that some of the permutations are not ideal distribution of training and test sets.

In Figure 3, similar to the detailed SARI score for **Wikipediax**, there are several divergences on certain permutations of **PubMedx**. In **PubMed11** and **PubMed16**, the average performance of expert models from **Wikipedia** outperforms the corresponding expert model. While in **PubMed6** and **PubMed9**, the average **Wikipedia** expert models show worse SARI scores than the base model. A similar inconsistency of the SARI score with the expected performance happens between the two generic models too. Considering the big performance gap between the GPT-3 and ChatGPT, the lack of more reference sentences may be one reason. The inconsistency of the detailed SARI score also shows the necessity of repeated experiments.

Comparing the base models with the generic

models, it is unclear why the generic models perform so poorly on the test sets in terms of the SARI score. One possible reason is that both base models and expert models are fitted with the optimal value of control tokens to maximize the SARI score while the prompted generic models are not. The calculation method of the SARI score prefers the sentence that keeps the most of original content under the condition of lack of reference sentences.

4.4 Case study

Table 5 shows the picked examples from the system. In the first example, ChatGPT demonstrates the ability of abbreviation explanation for the PoCT, while the other only follows the original text. In the second example, ChatGPT generates an inaccurate text that simplify the domestic animals as pets, which raises some concerns about the factuality of the simplification. In the third example, the generic models even removed the explanation of the abbreviation, which potentially decreased the readability of the sentence. The inconsistency of the performance of generic models can be an obstacle to applying such models to downstream tasks. In addition to that, the definition of simplicity for the generic models is also vague. We found that some of the outputs of ChatGPT are much shorter than the outputs of expert models. However, the short sentences don't always align with the simplicity and better readability.

4.5 Human evaluation

Table 6 shows the results of human evaluation. The scores range from 1 to 5, from strongly disagree to strongly agree. For the simplicity question, the generic model (ChatGPT) obtains a similar, but marginally higher score than the expert models under evaluation (**Wiki0** and **PubMed0**). However, for the meaning preservation question, ChatGPT was evaluated to have worse performance than the expert model. This implies that ChatGPT may have omitted some important details that the expert models were able to retain correctly. It also implies that the expert models retained much of the original text, making more conservative edits than the paraphrasing that was performed by GPT-3 and ChatGPT. Similar situations can also be found in Table 5 that the expert models tend to maintain the source content. Unlike the SARI score shown in Table 3 and 4, the performance gap between generic models and expert models is not as high as expected.

5 Discussion and Future Work

The performance of generic models is impressive in the general task. The generic model can become the new SOTA in many natural language processing tasks with proper prompts. However, the scale of the parameters in LLM like ChatGPT makes it almost impossible to be deployed locally. In addition, it can hardly be fine-tuned by an individual or a small group of researchers due to the high requirement for computation power. Even though it can be leveraged by prompts, when it comes to the specialised domain or private information, data privacy prevents it from becoming a universal solution for all people, which limits the applications in real-life scenarios. Another issue is low BERTScore in both general and genre-specific tasks, which indicates deviation in meaning preservation.

When it comes to genre-specific tasks, the generic model is less competitive than it is in general tasks. Based on the human evaluation (Table 6), the expert model shows similar or higher performance than the generic model. Although the generic model trained with a much larger corpus contains more internal knowledge, it is hard to determine where to stop the simplification without quantitive prompts. Considering the lower BERTScore from the generic models, it seems that the ChatGPT over-paraphrased the input, compared to the expert models. Results from human evaluation also agree that the expert models appear to preserve the meaning better than the generic model. However, the expert models are equipped with control tokens and inevitably will delete some content or information based on the control token they are given as well. This result shows that human annotators may prefer the more conservative settings of the expert models as opposed to the paraphrasing of the generic models in terms of meaning preservation.

The results in Table 3 and 4 prove the effectiveness of transfer learning after fine-tuning. As mentioned in the results, both expert models benefit from transfer learning in the genre-specific task. One possible reason for the improvement of the other kind of expert models may be the common ground between the two subsets of different genres, the topic and context for example. Yet the performance gap between the two types of expert models shows that genre or text style still matters and causes performance differences. The genre-adapted model can be a potential solution to better

Source:	For PoCT a single specimen of blood is usually obtained by skin puncture.
Reference:	For PoCT, a single specimen of blood is usually obtained by skin puncture.
GPT-3:	A skin puncture usually gets one sample of blood for PoCT.
ChatGPT:	For Point of Care Testing (PoCT), a small sample of blood is usually taken by pricking the skin.
Expert:	For PoCT, a single specimen of blood is usually obtained through skin puncture.
Source:	It is possible that many mammals including domestic animals are susceptible to SARS-CoV-2.
Reference:	It is possible that many mammals, including domestic animals, are susceptible to SARS-CoV-2.
GPT-3:	Mammals, including domestic animals, may be vulnerable to SARS-CoV-2.
ChatGPT:	Many types of animals, including pets, may be able to get infected with SARS-CoV-2.
Expert:	Many mammals including domestic animals may be susceptible to SARS-CoV-2.
Source:	Personal protective equipment (PPE) is considered less effective than engineering and administrative controls, but can help prevent some exposures.
Reference:	Personal protective equipment (PPE) is considered less effective than engineering and administrative controls. However, it can help prevent some exposures.
GPT-3:	PPE is not as good as engineering and admin controls but can prevent some exposures.
ChatGPT:	PPE is not as good as other ways to prevent exposure, but it can help some.
Expert:	Personal protective equipment (PPE) is considered less effective than engineering and administrative controls, but can help prevent some exposures.

Table 5: Examples of simplifications from different models

Model	Simplicity	Meaning Preservation
Generic	3.55	3.86
Expert	3.46	4.17

Table 6: Human evaluation score on test set of **Wiki0** and **PubMed0** (out of 5)

fit the requirements of different groups of lay users.

Even with highly capable generic or expert models, there is still the possibility for the introduction of factual errors in the output. With the convincing performance of generic models like ChatGPT, the hallucination problem become more serious than ever before. When the task is related to a crucial area such as medicine or legal help, the introduction of misleading information may cause severe problems. To improve the robustness of the simplification system, it is necessary to build a factual evaluation system in the future (Devaraj et al., 2022; Ma et al., 2022). Unlike other text generation tasks, simplification maintains the essential information in the input, thus it is easier to judge whether there is misleading content or hallucinations. BERTScore, which measures the meaning preservation for the implications, could be extended into a tool to measure the deviation of original meanings in future work.

Another problem is the explanation of abbreviations. For lay users unfamiliar with the abbreviations and technical terms, it is important to explain the meaning of these unique words or phases. ChatGPT has a huge knowledge base to understand common abbreviations. However, technical terms in certain domains may be unknown for the generic model and the abbreviations may refer to different phrases in different contexts. To avoid the above problem, the model needs to have a genre-specific knowledge base in future work, which allows the model to identify and explain the abbreviations and terms. To achieve this goal, a model competitive with an external source of knowledge base is required. In addition, the knowledge base should be combined with lexical complexity evaluation to decide which term needs explanation.

6 Conclusion

In this paper, we compared the performance differences between generic models and expert models on general and genre-specific simplification datasets. We showed the effect and practicality of transfer learning in genre-specific datasets with less amount of samples. The performance drop on general tasks after transfer learning is acceptable and may be further reduced in future studies. The performance, cost-effectiveness and portability of expert models prove themselves as one of the practical solutions for domains-specific or genres-specific tasks.

7 Lay Summary

Text simplification is a technique for making written language easier to read. This is helpful for people with reading difficulties such as dyslexia, or people who are learning a language. In this paper, we investigated how well tools built to simplify one type of text can be used to simplify another type of text. The two types of text we looked at were

academic articles and Wikipedia articles. To make these types of articles easier to read, we used large language models (such as ChatGPT), which were not designed for the task. Large Language Models are a new type of technology that are trained to complete a sentence, or write an appropriate response to a question. Language models are usually trained on general purpose data, so might not be useful for specialist areas such as academic articles and Wikipedia articles. We also designed our own customised models which were smaller, but trained on data that helped them to learn the task. As a result, we found that:

- the type of text (known as its genre) **does affect the performance** of text simplification models targeting general corpus;
- the **zero-shot large language models are competitive** but require tweaks to reach the same level of performance as the customized models;
- the smaller customized models may **still hold their position as the best model**.

References

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020a. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020b. Data-driven sentence simplification:: Survey and benchmark. *Computational Linguistics*, 46(1):135–187.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2021. The (Un)Suitability of Automatic Evaluation Metrics for Text Simplification. *Computational Linguistics*, 47(4):861–889.

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Franscisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. TICO-19: the translation initiative for COvid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Lucas Høyberg Puvis de Chavannes, Mads Guldborg Kjeldgaard Kongsbak, Timmie Rantzau, and Leon Derczynski. 2021. Hyperparameter power impact in transformer language model training. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 96–118, Virtual. Association for Computational Linguistics.

Will Coster and David Kauchak. 2011. Learning to simplify sentences using Wikipedia. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 1–9, Portland, Oregon. Association for Computational Linguistics.

Ashwin Devaraj, William Sheffield, Byron Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7331–7345, Dublin, Ireland. Association for Computational Linguistics.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *CoRR*, abs/2101.03961.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

David Kauchak. 2013. Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), pages 1537–1546, Sofia, Bulgaria. Association for Computational Linguistics.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338, Austin, Texas. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Xinyu Lu, Jipeng Qiang, Yun Li, Yunhao Yuan, and Yi Zhu. 2021. An unsupervised method for building sentence simplification corpora in multiple languages. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 227–237, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuan Ma, Sandaru Seneviratne, and Elena Daskalaki. 2022. Improving text simplification with factuality error detection. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 173–178, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Mounica Maddela, Fernando Alva-Manchego, and Wei Xu. 2021. Controllable text simplification with explicit paraphrasing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3536–3553, Online. Association for Computational Linguistics.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2020. Multilingual unsupervised sentence simplification. *CoRR*, abs/2005.00352.

Louis Martin, Benoît Sagot, Éric de la Clergerie, and Antoine Bordes. 2019. Controllable sentence simplification. *CoRR*, abs/1910.02677.

Daiki Nishihara, Tomoyuki Kajiwara, and Yuki Arase. 2019. Controllable text simplification with lexical constraint loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 260–266, Florence, Italy. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text simplification by tagging. *CoRR*, abs/2103.05070.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

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

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

J. Rapin and O. Teytaud. 2018. Nevergrad - A gradient-free optimization platform. `https://GitHub.com/FacebookResearch/Nevergrad`.

Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 712–718, Melbourne, Australia. Association for Computational Linguistics.

Lane Schwartz, Timothy Anderson, Jeremy Gwinnup, and Katherine Young. 2014. Machine translation and monolingual postediting: The AFRL WMT-14 system. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 186–194, Baltimore, Maryland, USA. Association for Computational Linguistics.

Thomas Scialom, Louis Martin, Jacopo Staiano, Éric Villemonte de la Clergerie, and Benoît Sagot. 2021. Rethinking automatic evaluation in sentence simplification. *CoRR*, abs/2104.07560.

Matthew Shardlow and Fernando Alva-Manchego. 2022. Simple TICO-19: A dataset for joint translation and simplification of COVID-19 texts. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3093–3102, Marseille, France. European Language Resources Association.

Kim Cheng Sheang and Horacio Saggion. 2021. Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of the*

14th International Conference on Natural Language Generation, pages 341–352, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Sanja Štajner, Kim Cheng Sheang, and Horacio Saggion. 2022. Sentence simplification capabilities of transfer-based models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(11):12172–12180.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 409–420, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Sander Wubben, Antal van den Bosch, and Emiel Krahmer. 2012. Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1015–1024, Jeju Island, Korea. Association for Computational Linguistics.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361.

A Template of human evaluation form

Please mark the score of simplicity and meaning preservation on a 5-point Likert scale. There are 2 sets of simplified sentences, please compare and mark the score.
Meaning preservation: To what extent do you agree the simplified sentence keeps the important information?
Simplisity: To what extent do you agree the simplified sentence is easy to understand?

Original Text	Simplified sentences	Meaning Preservation	Simplicity
It is possible that many mammals including domestic animals are susceptible to SARS-CoV-2.	Many types of animals, including pets, may be able to get infected with SARS-CoV-2.	Disagree	Agree
	Many mammals including domestic animals may be susceptible to SARS-CoV-2.	Agree	Agree

Table 7: Sample of the human evaluation form

B Detailed Scores

SARI score / Model \ Test set		Wiki0	Wiki1	Wiki2	Wiki3	Wiki4	Wiki5	Wiki6	Wiki7	Wiki8	Wiki9	Wiki10	Wiki11	Wiki12	Wiki13	Wiki14	Wiki15	Wiki16	Wiki17	Wiki18	Wiki19	**Overall**
Base	BART-base	39.58	43.13	40.20	40.82	40.99	40.67	39.33	40.14	40.53	40.88	42.18	42.43	43.30	41.17	37.74	41.03	41.99	39.70	39.34	40.48	40.78
Generic	GPT-3	29.54	27.85	29.82	28.29	28.62	29.70	28.41	28.22	29.12	30.22	27.83	30.01	29.76	29.18	28.88	28.62	28.44	30.68	28.42	28.95	29.03
	ChatGpt	32.46	31.05	32.16	30.45	32.07	29.63	30.21	29.93	31.73	31.79	29.21	31.83	31.62	29.85	29.16	32.04	31.37	32.86	31.92	31.07	31.12
Expert	Corresponding **Wiki** model	43.99	44.53	42.56	43.33	43.30	44.88	43.83	45.36	44.33	45.23	45.24	43.22	46.46	44.62	44.25	43.56	44.41	42.74	43.76	46.39	**44.30**
	PubMed0	43.52	43.36	42.62	44.94	44.33	41.31	41.79	42.81	42.02	43.25	41.68	41.51	44.38	43.41	42.90	43.36	43.03	41.57	45.32	44.14	43.06
	PubMed1	41.28	42.73	42.70	41.37	44.00	42.15	43.62	40.72	42.36	43.18	44.35	43.65	43.34	43.23	41.94	42.69	42.15	42.33	41.42	40.74	42.50
	PubMed2	42.72	43.78	43.01	43.16	41.49	44.52	40.69	41.39	42.03	42.19	40.90	43.48	45.08	42.52	41.60	43.95	44.92	40.46	43.88	41.42	42.66
	PubMed3	40.90	43.55	43.09	42.51	44.50	42.13	43.98	41.93	41.92	41.23	42.43	43.70	43.71	42.74	41.82	42.96	45.02	39.59	42.79	43.29	42.69
	PubMed4	40.04	43.34	44.72	43.26	43.14	43.85	43.09	40.27	41.62	44.31	41.43	42.11	42.56	42.43	42.25	41.42	43.33	41.50	41.79	42.54	42.45
	PubMed5	40.53	43.62	44.39	40.24	41.17	43.00	41.95	43.09	42.73	43.81	41.05	44.11	42.80	43.42	42.48	43.08	44.77	41.62	40.68	42.69	42.56
	PubMed6	43.11	42.98	43.84	42.94	40.59	42.03	43.91	40.63	40.95	42.81	44.03	43.80	43.05	42.59	41.02	43.49	44.00	42.00	42.28	42.95	42.65
	PubMed7	41.15	42.38	42.30	43.87	43.19	40.27	42.03	43.76	40.98	43.93	44.08	42.72	44.52	41.97	42.32	42.48	43.38	42.22	45.01	40.97	42.68
	PubMed8	43.59	43.58	41.92	43.81	42.54	42.74	44.77	41.77	42.25	43.69	40.03	43.96	44.83	42.11	41.88	44.40	44.89	41.48	42.48	43.39	43.01
	PubMed9	42.51	41.79	45.22	42.24	41.55	41.35	42.12	42.41	42.72	41.80	43.03	43.69	44.03	43.58	40.82	41.90	42.43	41.87	41.66	42.18	42.45
	PubMed10	42.11	44.12	45.41	42.17	42.80	41.11	42.83	43.29	42.80	41.20	43.50	43.33	42.80	43.29	42.55	42.22	43.43	41.15	42.33	42.58	42.75
	PubMed11	43.47	44.30	44.72	43.85	43.13	42.05	43.93	43.40	40.71	42.67	44.10	43.44	43.38	39.46	42.08	42.87	42.54	42.17	44.31	43.42	43.00
	PubMed12	42.17	44.73	43.65	43.23	42.29	40.96	41.82	41.15	43.31	44.12	42.38	44.00	42.25	40.82	41.67	43.07	43.19	40.36	42.72	40.23	42.41
	PubMed13	40.95	41.53	42.46	41.70	42.44	43.36	41.62	43.52	41.59	43.72	43.18	44.42	45.20	41.20	42.96	42.90	43.84	42.39	42.88	45.04	42.84
	PubMed14	41.49	43.38	44.17	42.49	42.83	42.85	44.60	42.53	41.09	44.83	43.16	42.87	44.32	42.02	43.41	43.98	44.91	43.21	44.16	41.56	43.19
	PubMed15	42.00	43.81	43.83	42.49	41.53	42.69	41.97	41.16	44.18	40.93	41.33	44.95	43.23	42.90	41.28	41.99	44.42	41.84	44.49	42.35	42.67
	PubMed16	43.38	41.63	42.97	42.11	41.39	42.59	42.32	43.22	42.85	45.12	41.27	42.06	44.34	43.17	41.33	42.46	44.40	42.24	44.35	39.71	42.65
	PubMed17	40.74	43.04	41.75	41.40	41.53	44.47	42.83	38.51	43.79	43.70	41.13	43.03	42.70	43.21	43.61	43.85	41.05	41.35	45.01	41.20	42.40
	PubMed18	42.27	43.16	45.59	41.33	44.20	41.83	43.75	42.24	41.86	43.91	41.39	43.87	44.77	44.06	41.01	43.57	43.79	41.16	44.14	43.23	43.06
	PubMed19	43.03	43.61	42.38	40.75	44.17	42.53	43.99	42.73	42.33	43.94	42.24	43.39	43.03	42.67	42.67	43.00	43.83	42.96	45.40	41.66	43.02
	PubMed20	42.67	42.60	42.72	42.02	42.76	42.95	43.42	42.08	43.34	39.79	43.97	42.78	44.28	42.11	41.81	42.49	43.82	42.77	42.26	41.22	42.59
	PubMed21	39.71	42.36	43.71	40.99	41.71	41.45	43.12	42.68	42.69	43.52	44.16	44.25	44.72	42.29	39.49	45.07	42.72	42.25	44.75	40.79	42.62
	PubMed22	42.09	41.96	42.83	41.27	41.09	43.91	42.91	41.61	42.06	42.85	42.64	42.87	44.37	43.12	42.54	43.15	42.30	42.59	43.21	41.46	42.54
	PubMed23	42.96	45.08	44.38	44.48	40.52	40.83	44.83	42.74	43.85	44.42	43.57	42.65	44.62	43.88	41.23	42.52	42.42	42.45	42.81	41.99	43.11
	PubMed24	43.37	43.43	43.09	40.50	40.55	42.57	43.25	41.98	43.70	43.95	43.88	43.19	45.14	42.82	42.61	43.41	44.45	42.12	44.37	41.20	42.98
	PubMed25	42.62	42.82	43.36	41.35	42.47	44.20	44.17	42.58	42.96	44.46	44.03	43.38	42.76	42.41	40.80	41.88	44.23	42.25	44.01	43.94	43.03
	PubMed26	42.76	44.63	42.64	43.09	42.91	42.57	43.57	42.71	41.95	42.63	44.82	42.51	42.08	41.11	40.34	43.53	44.59	41.85	44.50	41.92	42.84
	PubMed27	41.20	45.43	41.81	42.15	41.46	41.58	43.78	39.74	43.47	41.50	42.22	43.24	43.07	42.06	43.63	43.27	43.10	41.66	43.36	41.43	42.46
	PubMed28	39.86	41.25	43.70	44.96	42.41	41.11	42.69	43.74	40.87	42.40	41.69	44.37	44.64	41.88	43.59	42.84	44.57	41.20	44.66	41.58	42.70
	PubMed29	41.65	44.38	42.92	42.73	43.82	41.81	42.48	42.51	42.31	44.10	41.01	44.26	43.33	43.17	42.54	43.37	44.67	40.71	42.29	42.17	42.81
Average of **PubMed**		44.48	43.28	43.40	42.45	42.42	42.36	43.06	42.10	42.38	43.13	42.62	43.39	43.78	42.52	42.01	43.04	43.67	41.78	43.44	42.10	42.75

Table 8: Detailed SARI score on **Wikipedia*x*** (We use **Wiki** to refer the **Wikipedia** in the columns)

BERTScore / Model \ Test set		Wiki0	Wiki1	Wiki2	Wiki3	Wiki4	Wiki5	Wiki6	Wiki7	Wiki8	Wiki9	Wiki10	Wiki11	Wiki12	Wiki13	Wiki14	Wiki15	Wiki16	Wiki17	Wiki18	Wiki19	**Overall**
Base	BART-base	0.743	0.797	0.723	0.722	0.755	0.804	0.721	0.766	0.747	0.781	0.737	0.735	0.753	0.698	0.733	0.760	0.736	0.768	0.602	0.733	0.741
Generic	GPT-3	0.524	0.533	0.541	0.532	0.515	0.557	0.545	0.524	0.523	0.508	0.525	0.518	0.514	0.506	0.536	0.546	0.533	0.548	0.535	0.533	0.530
	ChatGpt	0.559	0.568	0.551	0.547	0.549	0.546	0.555	0.549	0.542	0.521	0.521	0.531	0.552	0.495	0.539	0.552	0.534	0.538	0.548	0.538	0.542
Expert	Corresponding **Wiki** model	0.763	0.756	0.805	0.736	0.745	0.772	0.764	0.771	0.738	0.796	0.749	0.748	0.774	0.727	0.770	0.726	0.711	0.755	0.749	0.763	**0.756**
	PubMed0	0.759	0.742	0.786	0.755	0.746	0.735	0.710	0.741	0.774	0.754	0.700	0.792	0.769	0.717	0.748	0.731	0.742	0.755	0.761	0.741	0.748
	PubMed1	0.669	0.771	0.776	0.731	0.746	0.724	0.781	0.736	0.745	0.784	0.794	0.733	0.772	0.713	0.785	0.744	0.764	0.715	0.786	0.708	0.749
	PubMed2	0.740	0.780	0.760	0.706	0.763	0.754	0.687	0.788	0.744	0.765	0.749	0.751	0.745	0.722	0.673	0.773	0.763	0.767	0.783	0.685	0.745
	PubMed3	0.670	0.781	0.750	0.680	0.747	0.758	0.810	0.769	0.753	0.730	0.756	0.769	0.715	0.710	0.646	0.703	0.759	0.673	0.675	0.729	0.729
	PubMed4	0.647	0.797	0.756	0.776	0.766	0.790	0.779	0.788	0.735	0.759	0.752	0.661	0.727	0.736	0.743	0.727	0.742	0.729	0.732	0.725	0.743
	PubMed5	0.649	0.799	0.736	0.676	0.704	0.763	0.763	0.753	0.762	0.756	0.737	0.737	0.720	0.726	0.743	0.660	0.750	0.735	0.745	0.705	0.731
	PubMed6	0.736	0.748	0.740	0.736	0.688	0.756	0.781	0.696	0.679	0.740	0.764	0.738	0.692	0.728	0.685	0.725	0.752	0.759	0.768	0.757	0.733
	PubMed7	0.734	0.769	0.740	0.788	0.751	0.745	0.756	0.764	0.730	0.761	0.786	0.718	0.745	0.720	0.759	0.691	0.759	0.727	0.771	0.705	0.746
	PubMed8	0.758	0.761	0.795	0.745	0.745	0.757	0.775	0.775	0.701	0.770	0.721	0.768	0.769	0.741	0.723	0.731	0.747	0.740	0.748	0.739	0.750
	PubMed9	0.752	0.811	0.758	0.723	0.778	0.719	0.780	0.758	0.766	0.683	0.751	0.742	0.730	0.736	0.707	0.707	0.747	0.754	0.787	0.709	0.745
	PubMed10	0.735	0.770	0.756	0.730	0.741	0.615	0.810	0.751	0.769	0.682	0.763	0.758	0.711	0.715	0.712	0.732	0.769	0.702	0.754	0.726	0.735
	PubMed11	0.738	0.766	0.773	0.786	0.750	0.762	0.782	0.769	0.660	0.714	0.781	0.759	0.767	0.574	0.761	0.729	0.724	0.762	0.763	0.739	0.743
	PubMed12	0.733	0.770	0.758	0.750	0.729	0.719	0.773	0.760	0.753	0.752	0.746	0.758	0.699	0.615	0.688	0.742	0.755	0.696	0.771	0.692	0.733
	PubMed13	0.726	0.673	0.748	0.713	0.732	0.773	0.726	0.759	0.755	0.756	0.770	0.705	0.764	0.685	0.762	0.713	0.746	0.735	0.754	0.739	0.737
	PubMed14	0.735	0.757	0.747	0.707	0.752	0.773	0.798	0.793	0.711	0.771	0.791	0.783	0.774	0.711	0.753	0.738	0.751	0.726	0.760	0.691	0.751
	PubMed15	0.695	0.784	0.765	0.735	0.703	0.746	0.741	0.726	0.771	0.782	0.715	0.743	0.728	0.737	0.762	0.705	0.783	0.761	0.782	0.727	0.745
	PubMed16	0.756	0.723	0.791	0.732	0.683	0.762	0.702	0.763	0.757	0.758	0.700	0.779	0.750	0.746	0.680	0.683	0.786	0.747	0.771	0.686	0.738
	PubMed17	0.654	0.771	0.701	0.640	0.753	0.750	0.785	0.625	0.758	0.774	0.714	0.743	0.740	0.720	0.752	0.728	0.685	0.725	0.749	0.701	0.723
	PubMed18	0.709	0.757	0.763	0.706	0.757	0.794	0.795	0.758	0.684	0.772	0.752	0.745	0.732	0.748	0.734	0.759	0.742	0.742	0.790	0.726	0.748
	PubMed19	0.750	0.776	0.791	0.692	0.762	0.773	0.793	0.748	0.723	0.755	0.777	0.770	0.698	0.728	0.768	0.720	0.727	0.748	0.768	0.688	0.748
	PubMed20	0.745	0.753	0.789	0.733	0.769	0.779	0.784	0.723	0.740	0.693	0.797	0.775	0.752	0.710	0.668	0.724	0.769	0.730	0.767	0.637	0.742
	PubMed21	0.723	0.812	0.743	0.693	0.761	0.707	0.798	0.748	0.724	0.735	0.800	0.748	0.753	0.713	0.701	0.740	0.711	0.729	0.765	0.681	0.739
	PubMed22	0.753	0.780	0.802	0.726	0.678	0.791	0.773	0.671	0.718	0.756	0.791	0.745	0.711	0.714	0.738	0.729	0.734	0.687	0.782	0.725	0.740
	PubMed23	0.734	0.796	0.774	0.770	0.676	0.698	0.780	0.776	0.766	0.735	0.788	0.739	0.757	0.751	0.707	0.743	0.774	0.759	0.768	0.720	0.750
	PubMed24	0.745	0.789	0.718	0.678	0.683	0.719	0.778	0.779	0.734	0.781	0.784	0.726	0.771	0.715	0.734	0.710	0.743	0.732	0.739	0.696	0.738
	PubMed25	0.768	0.780	0.749	0.753	0.740	0.787	0.784	0.766	0.709	0.759	0.806	0.744	0.757	0.756	0.731	0.714	0.772	0.739	0.765	0.749	0.756
	PubMed26	0.770	0.782	0.758	0.716	0.756	0.720	0.769	0.764	0.745	0.775	0.743	0.753	0.734	0.713	0.701	0.740	0.759	0.748	0.769	0.705	0.746
	PubMed27	0.626	0.790	0.697	0.705	0.762	0.723	0.779	0.632	0.770	0.774	0.756	0.749	0.695	0.744	0.744	0.729	0.744	0.753	0.761	0.691	0.731
	PubMed28	0.602	0.810	0.753	0.744	0.724	0.771	0.783	0.746	0.648	0.760	0.772	0.744	0.772	0.711	0.751	0.712	0.756	0.760	0.782	0.680	0.739
	PubMed29	0.630	0.779	0.749	0.734	0.742	0.772	0.762	0.774	0.751	0.783	0.683	0.733	0.715	0.744	0.742	0.727	0.748	0.745	0.646	0.719	0.734
Average of **PubMed**		0.715	0.773	0.757	0.725	0.736	0.748	0.771	0.747	0.735	0.752	0.758	0.747	0.739	0.717	0.727	0.724	0.750	0.736	0.759	0.711	0.741

Table 9: Detailed BERTScore on **Wikipedia*x*** (We use **Wiki** to refer the **Wikipedia** in the columns)

	SARI score / Model \ Test set	Pub0	Pub1	Pub2	Pub3	Pub4	Pub5	Pub6	Pub7	Pub8	Pub9	Pub10	Pub11	Pub12	Pub13	Pub14	Pub15	Pub16	Pub17	Pub18	Pub19	**Overall**
Base	BART-base	40.10	42.76	37.38	41.81	39.44	41.44	43.60	39.19	40.35	41.15	40.70	41.93	39.67	39.64	36.94	40.57	41.26	41.18	41.03	41.17	40.56
Generic	GPT-3	31.06	31.43	30.82	29.25	32.56	32.19	30.96	29.82	32.36	30.09	29.09	29.18	30.50	31.42	31.50	29.84	28.99	29.23	31.70	32.45	30.72
	ChatGpt	31.75	31.63	30.28	30.52	34.32	29.89	34.59	30.80	31.79	31.28	30.71	32.12	32.57	31.18	33.99	28.75	30.53	30.57	31.65	31.99	31.55
Expert	Corresponding **PubMed** model	46.04	44.02	41.77	45.95	44.83	43.98	43.60	43.61	43.45	44.97	47.79	42.92	48.14	46.05	44.45	45.97	42.66	46.41	48.24	46.11	**45.05**
	Wikipedia0	46.06	43.34	40.63	43.85	43.65	42.51	41.89	41.80	43.76	41.32	43.87	42.84	46.94	44.14	44.05	43.79	42.93	45.37	45.34	43.25	43.57
	Wikipedia1	43.39	42.57	40.29	45.24	44.20	45.11	43.38	41.63	41.96	40.52	45.66	42.03	45.38	44.98	43.79	43.27	44.64	45.83	46.75	42.95	43.68
	Wikipedia2	45.31	43.26	40.92	45.10	42.84	44.82	42.80	40.55	43.14	42.57	44.47	43.55	42.91	43.91	42.59	44.53	42.33	44.46	43.98	43.53	43.38
	Wikipedia3	44.41	42.11	40.26	43.42	43.47	44.51	42.58	42.72	42.73	39.74	43.53	43.33	46.24	42.25	43.68	43.69	41.94	46.19	41.94	44.19	43.15
	Wikipedia4	44.70	40.70	41.02	43.48	42.02	43.26	42.17	41.30	43.34	39.49	43.39	44.76	46.35	42.92	41.51	46.17	40.74	46.15	46.13	42.63	43.11
	Wikipedia5	46.16	41.24	40.83	44.85	41.77	42.62	41.47	41.97	42.22	41.37	44.00	43.46	44.24	41.83	41.15	45.41	42.23	47.27	44.74	44.21	43.15
	Wikipedia6	42.23	44.98	40.99	44.10	41.80	42.62	43.02	41.68	42.84	40.82	45.20	44.38	45.67	41.93	42.06	45.43	43.81	45.50	46.08	44.49	43.48
	Wikipedia7	44.93	44.30	40.96	43.90	44.42	44.90	40.82	43.56	42.10	40.39	42.95	43.19	43.69	43.02	41.53	44.58	42.79	43.82	45.91	43.77	43.28
	Wikipedia8	43.85	42.81	41.95	43.02	42.90	41.77	39.76	42.84	41.78	40.11	45.05	44.83	45.78	43.62	42.97	45.73	43.42	44.07	46.07	44.24	43.33
	Wikipedia9	45.88	42.68	41.28	44.54	45.69	43.88	43.34	42.39	42.65	41.62	44.99	45.20	44.93	42.61	42.95	43.88	42.75	43.72	44.58	42.70	43.61
	Wikipedia10	45.14	42.52	40.63	45.48	43.21	45.29	43.57	43.63	43.68	42.19	44.79	44.51	42.59	41.15	40.73	44.78	45.95	44.84	45.45	43.80	43.70
	Wikipedia11	43.05	44.73	42.57	44.13	42.20	42.16	43.28	42.06	43.18	39.55	45.14	44.82	47.42	41.86	44.28	45.07	42.27	44.12	44.45	44.25	43.53
	Wikipedia12	43.74	42.67	41.42	45.30	41.43	41.50	41.39	41.15	43.90	40.54	42.88	44.52	46.35	41.43	42.16	44.76	44.87	46.92	43.65	43.51	43.20
	Wikipedia13	42.95	42.03	41.47	43.44	45.47	42.90	41.41	41.84	42.90	39.49	40.63	42.90	43.59	42.72	41.57	43.80	43.22	46.46	45.58	43.94	42.92
	Wikipedia14	41.49	44.27	42.48	44.06	42.46	43.58	42.98	43.13	42.39	40.88	44.66	42.15	42.47	42.40	44.49	44.27	42.60	45.02	41.65	44.04	43.07
	Wikipedia15	44.64	44.71	40.12	44.81	43.36	42.47	42.25	43.47	42.77	39.17	42.31	44.99	45.34	40.44	43.39	44.06	42.94	45.17	44.63	43.21	43.21
	Wikipedia16	43.01	45.84	41.05	44.86	42.96	44.49	43.00	41.09	43.01	39.56	44.48	43.54	44.58	44.12	42.18	45.60	39.94	45.22	44.71	43.64	43.34
	Wikipedia17	43.33	42.32	40.62	44.18	44.55	43.13	42.40	41.62	43.21	39.36	44.05	41.84	43.14	43.73	44.08	44.30	40.57	44.06	47.49	43.21	43.06
	Wikipedia18	44.74	44.14	40.04	45.04	42.26	41.52	44.70	42.95	43.10	39.69	44.52	42.64	42.61	41.88	41.25	46.20	45.05	45.41	44.07	41.83	43.18
	Wikipedia19	44.72	44.90	40.79	43.83	43.62	44.99	41.02	42.08	43.39	41.20	41.86	43.90	46.57	43.21	44.47	46.07	43.44	46.22	45.68	43.80	43.79
	Wikipedia20	45.78	45.79	39.14	44.39	41.65	44.54	40.85	43.16	42.07	40.49	43.16	43.40	45.53	42.08	42.91	44.46	42.23	43.54	46.00	44.26	43.27
	Wikipedia21	42.83	45.37	42.21	43.96	41.52	44.55	42.12	42.13	43.92	41.05	43.88	42.76	46.10	42.38	40.89	44.05	42.10	46.18	42.55	43.93	43.22
	Wikipedia22	43.98	43.17	41.61	43.62	43.51	43.34	41.60	42.12	42.86	39.97	44.90	43.50	44.82	44.72	43.07	44.65	42.30	44.55	43.78	42.76	43.24
	Wikipedia23	45.41	42.82	40.86	43.01	42.46	42.87	42.14	41.89	45.22	40.58	41.75	43.27	43.70	42.96	43.21	45.18	45.62	47.94	44.51	42.69	43.40
	Wikipedia24	45.29	43.69	41.22	45.13	45.94	45.19	43.28	42.91	44.35	42.44	43.45	43.45	44.02	42.62	44.39	44.78	43.45	47.87	44.89	43.92	44.11
	Wikipedia25	45.31	44.64	43.61	46.07	40.68	42.70	42.50	42.01	44.01	41.90	46.28	45.02	43.01	41.03	43.60	44.47	43.90	42.92	45.81	43.83	43.66
	Wikipedia26	45.61	42.98	39.72	44.76	43.94	42.53	40.98	42.57	43.08	41.54	42.74	41.86	44.59	41.40	43.18	45.23	44.35	43.14	41.99	42.43	42.93
	Wikipedia27	45.89	44.58	41.34	44.53	44.02	41.21	41.74	41.48	42.74	41.51	45.32	43.03	47.85	43.81	41.63	45.04	43.74	48.39	45.90	43.68	43.87
	Wikipedia28	45.80	42.73	40.32	43.46	44.22	41.83	41.92	42.16	41.86	41.03	43.89	43.94	43.00	44.00	41.16	45.65	40.53	46.38	43.78	42.17	42.99
	Wikipedia29	44.68	46.07	40.86	43.56	43.52	45.32	43.29	42.78	42.91	41.47	44.80	44.57	43.20	43.57	43.92	45.95	42.82	43.40	45.21	45.02	43.85
Average of **Wikipedia**		44.48	43.60	41.04	44.30	43.19	43.40	42.26	42.22	43.04	40.72	43.95	43.61	44.75	42.76	42.76	44.83	42.98	45.34	44.78	43.53	43.38

Table 10: Detailed SARI score on **PubMed*x***(We use **Pub** to refer the **PubMed** in the columns)

BERTScore / Test set / Model		Pub0	Pub1	Pub2	Pub3	Pub4	Pub5	Pub6	Pub7	Pub8	Pub9	Pub10	Pub11	Pub12	Pub13	Pub14	Pub15	Pub16	Pub17	Pub18	Pub19	**Overall**
Base	BART-base	0.739	0.773	0.691	0.725	0.680	0.702	0.735	0.732	0.681	0.756	0.720	0.778	0.646	0.728	0.710	0.741	0.725	0.730	0.731	0.743	0.723
Generic	GPT-3	0.542	0.557	0.549	0.536	0.546	0.565	0.538	0.509	0.550	0.525	0.545	0.549	0.568	0.543	0.519	0.573	0.561	0.572	0.540	0.556	0.547
	ChatGpt	0.517	0.537	0.505	0.511	0.535	0.497	0.527	0.507	0.513	0.509	0.510	0.511	0.508	0.501	0.521	0.512	0.533	0.527	0.491	0.521	0.515
Expert	Corresponding **PubMed** model	0.767	0.774	0.675	0.749	0.725	0.705	0.751	0.748	0.724	0.739	0.758	0.711	0.782	0.768	0.737	0.782	0.701	0.770	0.776	0.688	**0.741**
	Wikipedia0	0.742	0.743	0.723	0.737	0.714	0.658	0.702	0.724	0.709	0.712	0.702	0.723	0.742	0.764	0.686	0.760	0.731	0.722	0.714	0.722	0.721
	Wikipedia1	0.683	0.761	0.713	0.746	0.742	0.755	0.704	0.650	0.720	0.772	0.745	0.681	0.711	0.748	0.670	0.781	0.726	0.750	0.721	0.659	0.722
	Wikipedia2	0.743	0.754	0.711	0.748	0.656	0.756	0.730	0.664	0.758	0.758	0.756	0.754	0.654	0.743	0.668	0.774	0.718	0.748	0.709	0.731	0.727
	Wikipedia3	0.748	0.729	0.720	0.709	0.723	0.750	0.743	0.724	0.712	0.757	0.698	0.710	0.778	0.718	0.647	0.741	0.710	0.745	0.689	0.739	0.725
	Wikipedia4	0.731	0.732	0.688	0.711	0.704	0.716	0.718	0.700	0.740	0.748	0.728	0.773	0.738	0.687	0.689	0.774	0.667	0.757	0.738	0.719	0.723
	Wikipedia5	0.769	0.745	0.704	0.729	0.700	0.696	0.732	0.676	0.732	0.744	0.749	0.771	0.754	0.730	0.684	0.800	0.741	0.762	0.762	0.744	0.736
	Wikipedia6	0.777	0.751	0.725	0.712	0.689	0.674	0.699	0.681	0.747	0.733	0.726	0.741	0.756	0.717	0.707	0.762	0.759	0.761	0.729	0.735	0.729
	Wikipedia7	0.744	0.746	0.650	0.712	0.760	0.761	0.719	0.704	0.697	0.743	0.687	0.707	0.713	0.757	0.672	0.778	0.732	0.758	0.728	0.723	0.725
	Wikipedia8	0.715	0.744	0.730	0.743	0.749	0.701	0.727	0.714	0.695	0.737	0.737	0.765	0.758	0.757	0.698	0.765	0.724	0.731	0.720	0.751	0.733
	Wikipedia9	0.760	0.731	0.688	0.712	0.754	0.731	0.748	0.715	0.746	0.768	0.748	0.794	0.735	0.734	0.680	0.790	0.722	0.626	0.685	0.733	0.730
	Wikipedia10	0.741	0.749	0.699	0.724	0.747	0.754	0.743	0.740	0.707	0.740	0.748	0.760	0.646	0.780	0.629	0.775	0.777	0.770	0.741	0.751	0.736
	Wikipedia11	0.723	0.725	0.734	0.713	0.656	0.724	0.725	0.725	0.655	0.713	0.725	0.755	0.778	0.756	0.691	0.769	0.746	0.759	0.715	0.713	0.725
	Wikipedia12	0.726	0.716	0.692	0.766	0.708	0.715	0.734	0.709	0.723	0.726	0.681	0.792	0.735	0.705	0.723	0.766	0.743	0.750	0.729	0.668	0.725
	Wikipedia13	0.784	0.786	0.700	0.713	0.749	0.711	0.714	0.647	0.755	0.764	0.638	0.771	0.692	0.743	0.664	0.727	0.728	0.754	0.729	0.737	0.725
	Wikipedia14	0.779	0.743	0.747	0.711	0.703	0.693	0.747	0.731	0.686	0.727	0.709	0.711	0.644	0.737	0.668	0.749	0.702	0.743	0.656	0.718	0.715
	Wikipedia15	0.769	0.783	0.684	0.713	0.710	0.699	0.716	0.690	0.723	0.741	0.695	0.803	0.749	0.676	0.705	0.745	0.743	0.765	0.708	0.742	0.728
	Wikipedia16	0.749	0.751	0.695	0.759	0.701	0.726	0.693	0.714	0.737	0.704	0.740	0.727	0.714	0.751	0.662	0.755	0.666	0.764	0.720	0.733	0.723
	Wikipedia17	0.761	0.764	0.700	0.703	0.735	0.711	0.746	0.681	0.737	0.759	0.698	0.685	0.664	0.733	0.665	0.762	0.682	0.734	0.729	0.716	0.718
	Wikipedia18	0.758	0.780	0.686	0.732	0.705	0.681	0.735	0.703	0.744	0.736	0.743	0.707	0.686	0.715	0.678	0.791	0.785	0.736	0.733	0.642	0.724
	Wikipedia19	0.733	0.748	0.681	0.735	0.644	0.750	0.675	0.692	0.727	0.690	0.674	0.704	0.745	0.726	0.684	0.799	0.736	0.747	0.705	0.739	0.717
	Wikipedia20	0.761	0.725	0.676	0.718	0.671	0.743	0.698	0.704	0.722	0.727	0.703	0.748	0.751	0.755	0.698	0.765	0.728	0.751	0.738	0.735	0.726
	Wikipedia21	0.743	0.759	0.735	0.691	0.648	0.731	0.742	0.697	0.774	0.729	0.755	0.697	0.763	0.715	0.700	0.745	0.711	0.765	0.672	0.752	0.726
	Wikipedia22	0.759	0.764	0.692	0.683	0.706	0.707	0.691	0.694	0.681	0.744	0.762	0.726	0.709	0.749	0.681	0.769	0.750	0.734	0.741	0.766	0.725
	Wikipedia23	0.753	0.752	0.701	0.719	0.661	0.733	0.703	0.688	0.763	0.718	0.705	0.749	0.716	0.756	0.641	0.794	0.790	0.743	0.720	0.738	0.727
	Wikipedia24	0.738	0.759	0.723	0.728	0.708	0.756	0.718	0.713	0.734	0.737	0.705	0.664	0.719	0.682	0.657	0.729	0.737	0.746	0.731	0.700	0.719
	Wikipedia25	0.731	0.740	0.743	0.770	0.766	0.688	0.740	0.702	0.737	0.657	0.743	0.737	0.719	0.703	0.730	0.779	0.718	0.716	0.732	0.726	0.729
	Wikipedia26	0.760	0.680	0.691	0.731	0.709	0.680	0.734	0.735	0.766	0.734	0.659	0.724	0.725	0.689	0.671	0.787	0.722	0.736	0.642	0.646	0.711
	Wikipedia27	0.734	0.742	0.694	0.718	0.739	0.728	0.737	0.655	0.718	0.734	0.741	0.685	0.777	0.783	0.685	0.774	0.756	0.769	0.719	0.721	0.730
	Wikipedia28	0.752	0.792	0.696	0.732	0.703	0.746	0.738	0.700	0.729	0.767	0.767	0.759	0.725	0.723	0.685	0.792	0.669	0.759	0.701	0.740	0.734
	Wikipedia29	0.765	0.756	0.678	0.741	0.712	0.768	0.711	0.715	0.722	0.729	0.747	0.757	0.746	0.766	0.723	0.777	0.698	0.750	0.743	0.722	0.736
Average of **Wikipedia**		0.748	0.748	0.703	0.725	0.709	0.721	0.722	0.700	0.727	0.735	0.721	0.736	0.725	0.733	0.681	0.769	0.727	0.745	0.717	0.722	0.726

Table 11: Detailed BERTScore on **PubMed*x***(We use **Pub** to refer the **PubMed** in the columns)