Enhancing Fact-Checking of Biomedical Literature Summarization

Maggie Xiong
McGill University
maggie.xiong@mail.mcgill.ca

Signe Hoel
McGill University
signe.hoel@mail.mcgill.ca

Abstract

Many modern summarization tools are prone to producing hallucinations. This means information provided in the summary is not directly inferable from the original text itself. To detect information that may be incorrectly produced by these models, fact-checking models have also been on the rise. While many novel fact-checking models exist, they mainly focus on the detection of non-factual information in news articles. The medical and biological sciences domain, another scope which relies heavily on factual consistency, remains relatively untouched when it comes to fine-tuning or training fact-checking models. Therefore, we use FactCC, a novel fact-checking model, and fine-tune it on the PubMed Dataset - a popular dataset pertaining to biomedical literature. To produce summaries that our FactCC model will use, we will use articles from the PudMed dataset as our source text and use FactCC's custom synthetic data generation functions to produce negative and noisy examples. Our findings provide three contributions: 1) The accuracy and fine-tuning protocol of the FactCC model on the PubMed dataset, 2) An analysis of the hallucinations potentially produced by the BERT model on the PudMed dataset and 3) Additional synthetic data generation mechanisms in addition to the current implementations provided by the original paper. We propose a new procedure for generating and fact-checking summaries for biomedical literature and terms using existing architecture. This can promote the use of information exchange within the medical domain that is both accurate and digestible.

1 Introduction

Summarization systems in models such as T5 (Raffel et al., 2020) and Bert (Devlin et al., 2019) are expanding in popularity due to their known accuracy and usage. However, recent studies have shown that these systems are prone to hallucinations - information produced within the summary that may not be contextually true to the given source text (Cao et al. 2022a, Sun et al. 2020, Tenney et al. 2020, Hu et al. 2020). Hu et al. (2020) even showed that 73.1 percent of summaries produced by the BERTS2S model on a random sample of 500 articles from the British Broadcasting Corporation (BBC) dataset showed some form of hallucination as quantified by human annotations.

Consequently, over time, fact-checking models have been introduced as well (Kryscinski et al. 2020, Cao et al. 2022b). Different methods for fact-checking are explored such as learning with rejection in the RejSumm model (Cao et al., 2022b), which was shown to improve the factuality of generated summaries compared to five baseline models, and factual consistency testing in FactCC (Kryscinski et al., 2020), which outperforms previously explored models such as BERT+MNLI and BERT+FEVER based on accuracy and F1-score.

However, we noticed that most popular fact-checking models are not fine-tuned or trained on biomedical or clinical datasets despite the interaction of text summarization on biomedical language being a popular discussion (Alkaissi and McFarlane 2023, Ahmad et al. 2023, Alambo et al. 2022). Given the sensitivity of information processing when it comes to the medical domain, we believe more fact-checking models should be fine-tuned or trained on these domain specific datasets, as these models can be beneficial in the medical domain to improve efficiency and patient communication.

Therefore, we aim to fine-tune the FactCC model (Kryscinski et al., 2020) on the well-known PudMed dataset (Cohan et al., 2018), a comprehensive repository on biomedical literature, in order to promote the use of factually correct medical summarizations generated by an uncased, base BERT architecture. This dataset comprises more than 36 million citations for biomedical literature and serves as a substantial dataset to train the FactCC model on. Meanwhile, the FactCC model, trained on the CNN/DailyMail dataset, aims to verify factual consistency and denote conflicts between documents and summaries generated by state-of-the-art summarization models.

The general procedure is as follows. Firstly, we fine-tuned the uncased, base BERT architecture used in the original paper on the PubMed dataset. From there, we preprocessed the summaries from the train, validation, and test split of the PubMed dataset into the corresponding JSON objects as indicated by the FactCC model. Then we used the FactCC's helper modules to generate synthetic claims that are positive, negative, and with noise as well. Once we had the generated data using the transformations supplied by the FactCC model, we fine-tuned the model on our training dataset for PubMed. Note that a random sample subset of the PubMed training dataset was used based on computational resources and memory constraints.

From there, we used weighted balanced accuracy, F1, and loss scores as seen in the original paper to see how well the model performs on the training data and also aggregate the binary classifications produced by our model to see if hallucinations were detected on the summaries produced by the BERT model. This aims to analyze the trustworthiness of the model on this domain-specific dataset. Additionally, the original code includes several methods for generating synthetic data by adding noise to the data such as pronoun swaps and negating the text. We implemented two new methods into the code that can be usable for future applications of the code.

Our research aims to address the gap in fact-checking when it comes to biomedical literature text summarization, but also examines the mitigation of hallucinations produced in biomedical text summarization by the BERT model. Our study aims to enhance insights into the reliability and accuracy of fact-checking models within a specialized domain. We envision that our findings can contribute to future work in utilizing text summarization to produce reliable and accurate information in the critical field of healthcare and medicine so everyone involved can benefit in the use of text summarization for biomedical information.

2 Related Works

Previous research, such as a study conducted by Cao et al. (2018), highlighted the prevalence of factual inconsistencies in abstractive summarization systems. Their findings indicated that approximately 30% of summaries generated by generative models contained at least one factual inconsistency. In scientific and medical domains, where precision and reliability are crucial, addressing these inconsistencies becomes crucial, especially when leveraging large language models (LLMs) for text summarization.

Various evaluation methods have been employed to measure hallucinations in generated summaries. Human evaluation methods, such as FactScore (Min et al., 2023), break down generations into atomic facts and rely on human evaluators to assess their accuracy. However, such methods can be resource-intensive and subjective.

Popular n-gram similarity metrics such as BLEU, ROUGE, and METEOR (Papineni et al. 2002, Lin 2004, Lavie and Agarwal 2007) directly measure the similarity between generated summaries and source text without constructing new resources. Unfortunately, these non-model based evaluation metrics exhibit low correlation with human judgements for factual consistency and are therefore unsatisfactory in this context. Therefore, researchers have focused on exploring more complex model-based architectures for both training and inference.

Recent works such as QAGS and QuestEval (Wang et al. 2020, Scialom et al. 2021) adopt question generation and question answering frameworks to evaluate factual consistency. Both methods first generate questions using entities or noun phrases in the candidate summary and then compare the answers of these questions between source and summary. Although these methods do have a high correlation with human judgements than previous metrics in consistency checking, they are computational complex and errors in each component can be cascaded.

Natural Language Inference (NLI) models, often trained on SNLI and MNLI datasets (Bowman et al. 2015, Williams et al. 2018), focus on classifying logical entailment between document and summary single-sentence pairs. However, applying these models to factual consistency checking in long-form, multi-sentence reasoning scenarios presents challenges. To address these challenges, studies such as DocNLI and FactCC (Yin et al. 2021, Kryscinski et al. 2020) propose a document-sentence approach, verifying each summary sentence against the entire source document rather than on a sentence-sentence level.

Currently, supervised training datasets for factual consistency are scarce and expensive to create. In the medical space, this is especially true. Both DocNLI and FactCC explore alternative approaches, including the synthesis of datasets using pre-defined rules such as entity substitution or mask-and-fill. These approaches provide viable alternatives to overcome the limitations of acquiring large-scale, high-quality datasets through human annotation (Kryscinski et al. 2020).

The interpretability of fact-checking models is crucial for understanding their decision-making processes, particularly in a clinical context. FactCC, for instance, introduces FactCCX, a version of the model equipped with additional span selection heads (Kryscinski et al., 2020). This enhancement allows the model not only to identify spans in the source document that support a claim but also to highlight spans in the claim where a potential mistake may have occurred. This interpretative capability enhances the transparency of fact-checking models.

The related work underscores the challenges of addressing factual inconsistencies in abstractive summarization systems, the limitations of current evaluation methods, and the ongoing efforts to enhance the interpretability of fact-checking models. Our study builds upon these insights to specifically tackle hallucinations in biomedical text summarization using the FactCC fact-checking model fine-tuned on the PubMed dataset. As FactCC/-FactCCX is well-documented, heavily-cited and offers interpretable fact-checking capabilities, it is a great candidate model for our purposes.

3 Modeling

3.1 Baseline Model

We have used the FactCC model as our baseline to serve as a point of comparison and expansion for our proposed model. This baseline model performs the task of classifying claims based on source texts as either factual or non-factual with direct supporting evidence provided.

The FactCC model employs a deep learning architecture, in both the original paper's and our case, this will be the pre-trained uncased, base BERT language model. The FactCC model is trained on the labeled CNN/DailyMail article dataset which includes non-fiction news articles and sources.

This proves as a trustworthy baseline model because it has demonstrated effectiveness in providing accurate classifications, as it outperforms BERT+MNLI and BERT+FEVER models in both weighted accuracy and F1-score. For example, BERT+MNLI and BERT+FEVER have weighted accuracies of 51.51 and 52.07 respectively while FactCC has a weighted accuracy of 74.15. In terms of F1-score, it boasts a relevant score of 0.5106 in comparison to the 0.0882 and 0.0857 F1-scores by the BERT+MNLI and BERT+FEVER models respectively. Additionally, it uses the same language model we planned to use for our own classification task, therefore, it sets out as a valuable comparison as a baseline to our new proposed model. Lastly, the documentation and code resources provided by the original authors make it easy to replicate and train for our own purposes. The ability to see and adapt to the source code is valuable with our specific goal in mind.

3.2 Our Custom FactCC

Our proposed model uses the FactCC architecture proposed for general datasets and is fine-tuned specifically for biomedical text summarization literature based on the PubMed dataset. Additionally, we also provide two new mechanisms for synthetic data generation in addition to the original ones presented in the code. This can be used in the future to enhance the learning process of the model for greater accuracy. We aim to enhance the reliability of text summarization within the medical domain by offering fact-checking mechanisms.

The modifications of the architecture of the original model to produce our new proposed model are discussed below.

3.2.1 Fine-tuning FactCC

The FactCC model is trained on the PubMed dataset to be fine-tuned towards biomedical literature and text. This allows the model to understand the complexities and inner workings of language and writing styles related to that specific domain.

3.2.2 Integration with BERT

Similar to the original model, we will leverage the BERT language model as the underlying architecture for FactCC to produce the summaries and capture the dependencies in the source text itself.

3.2.3 Additional Synthetic Data Generation Methods

FactCC's helper modules are used to generate synthetic claims that include positive, negative, and noisy examples. These allow the model to be trained on and learn from a diverse set of examples for greater accuracy. Therefore, we will add our own methods to the helper modules in order to provide more ways to generate claims to enhance the diversity of examples and textual scenarios.

There are many advantages proposed by this model. Our model will gain domain specific knowledge and expertise within the medical field and be able to understand the nuanced language within this field. This expands the diversity of the original model given it was not trained on this specific domain previously. Consequently, our model will mitigate hallucinations in the biomedical domain specifically as the use of this model can prevent hallucinations generated by language models such as BERT.

While FactCC performs well on a general case basis, our proposed model excels in the biomedical domain compared to the general FactCC model as it is tailored based on the PubMed dataset. The domain-specific knowledge and enhanced synthetic data generation mechanisms that can be used during training contribute to a trustworthy and accurate model for fact-checking biomedical text summarization.

4 Dataset and Evaluation

4.1 Data

The dataset that we used to fine-tune the BERT summarization model and train the FactCC fact-checking model will be the PubMed dataset (Cohan et al., 2018), accessed through the Huggingface datasets library. This dataset was originally obtained from the PubMed OpenAccess repository, and is already split up into training, validation and test set files. The files are in jsonline format with each json object in the following format:

```
{
    "abstract": "string",
    "article": "string",
    "section_names": "string"
}
```

Figure 1: Example of a JSON object in the PubMed OpenAccess repository

An example of a json object belonging to the ’train’ set looks as follows (cropped for brevity):

```
{
    "abstract": "\" we have
     ↪  studied the leptonic
     ↪  decay @xmath0 , via
     ↪  the decay channel
     ↪  @xmath1 , using a
     ↪  sample of tagged
     ↪  @xmath2 decays
     ↪  collected...",
    "article": "\"the
     ↪  leptonic decays of a
     ↪  charged pseudoscalar
     ↪  meson @xmath7 are
     ↪  processes of the type
     ↪  @xmath8 , where
     ↪  @xmath9 , @xmath10 ,
     ↪  or @...",
    "section_names":
     ↪  "[sec:introduction]
     ↪  introduction\n
     ↪  [sec:detector]data
     ↪  and the cleo-
     ↪  detector\n[sec:analysys]
     ↪  analysis
     ↪  method\n[sec:conclusion]
     ↪  summary"
}
```

Figure 2: Example of a JSON object belonging to the train set in the PubMed OpenAccess repository

The PubMed dataset was chosen as a suitable option for multiple reasons. Firstly, it contains diverse scientific content such as a range of disciplines, topics, and writing styles. This allows the model to learn from an extensive range of examples during the training process so it can better fit to unseen data when evaluated. Similarly, the PubMed dataset uses real life articles and examples which ensures that all information is realistic and challenges the model to learn to the extent of real-world scenarios. The PubMed dataset is also very well-structured. As denoted above, each json object represents an article with all subsequent information already being tokenized properly. This reduces the amount of preprocessing that has to be completed in order to use the data. Additionally, the dataset is already split properly into training, validation, and test sets. This allows for consistency during training and evaluation.

The size of the PubMed downloaded dataset files cumulates to 4.50 GB. The train, validation and test set have 119924, 6633 and 6658 examples respectively. For the validation set, the length of the articles ranges from 0 to 991348 characters; the length of the abstracts in that set range from 180 to 3180 characters, and the length of the section names in that set ranges from 1 to 1080 characters.

4.2 Metrics

To evaluate our custom trained FactCC model, we will use the same evaluation metrics as shown in the original paper - weighted (class-balanced) accuracy and F1-score. These metrics are used to evaluate our model's ability to correctly classify factual and non-factual claims. The equations for such are shown below.

4.2.1 F1-Score

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{2}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{3}$$

4.2.2 Weighted Balanced Accuracy

The weighted balanced equation as shown in Gupta et al. (2021).

$$\sum_{i=1}^{C} w_i \times Accuracy_i \tag{4}$$

where C is the number of classes, w_i is the weight assigned to class i and $Accuracy_i$ is the accuracy for class i.

5 Experimental Details

5.1 Fine-tuning Process

The general procedure was composed of three main parts.

5.1.1 Gathering the Data

The PubMed dataset had a training, validation, and test split already. Therefore, we extracted a subset of each of the datasets to use for fine-tuning and evaluation. Originally, the training dataset has 119,924 examples, the validation has 6,633 examples and the test dataset has 6,658 examples. However, acknowledging potential resource constraints detailed throughout this section, we decided to take a subset of 1000 training examples, 125 validation examples, and 125 test examples. We chose these specific numbers because the common split ratio would be 80% for training, 10% for validation, and 10% for testing. Therefore, we wanted to mimic a possible ratio split similar to that common ratio as closely as we could.

To create the subset examples, we ran a simple python script that iterated through and generated a random sample of the following data with the respective sizes. Each entry was saved as a json object with the following format Figure 3 in the respective jsonl files titled "data-$\%s$.jsonl" where $\%s$ corresponds to either 'train', 'valid' or 'test.

```
{
    "Id": "int",
    "Text": "str"
}
```

Figure 3: Example of a JSON object processed for fine-tuning.

5.1.2 Evaluating the Baseline Model

Before fine-tuning the model, we evaluated the baseline FactCC model against the validation and test set. As mentioned in the previous section, each set consisted of 125 examples. For synthetic data generation that is used for evaluation, the FactCC model uses various synthetic data generation methods as mentioned in the previous section. Therefore, using their helper scripts, we first generated a total of 600 examples including the original ones, where an additional 475 were generated with the transformations applied.

Following that, we created a helper module to combine the jsonl files formulated by the various transformations as each transformation generated their own files for both positive and negative examples with noise applied. Our helper module helped combine this into one jsonl file that we could then run for evaluation. The evaluation script was then run and the results were stored in their proper files.

Additionally, it should be noted that almost all of the libraries and software dependencies that we downloaded had to be changed in version in order to accommodate a more recent python version.

5.1.3 Fine-tuning the Baseline Model on PubMed

The model was fine-tuned on the PubMed dataset using the following parameters in Table 1 and used the following software dependencies in Table 2.

Fine-tuning Parameters	
Training Epochs	5
Number of Runs	3
Batch Size	12
Gradient Accumulation Steps	1
Total Optimization Steps	3700
Software Dependencies	
GPU	McGill GPU
Training Time	19 hours
Execution Method	tmux session
Code Repository	Repository

Table 1: Fine-tuning parameters and software dependencies for fine-tuning the baseline model on the PubMed dataset

Library	Version
python	3.10
pytorch	1.2.0
scipy	1.12.0
space	3.7.4
torch	1.3.1
numpy	1.26.4
tqdm	4.66.2
apex	0.9.10dev
protobuf	4.25.3
scikit_learn	1.4.1.post1
google-cloud-translate	1.6.0

Table 2: Libraries and their respective versions used during the fine-tuning and evaluation process

For fine-tuning with the PubMed dataset, there were computational restrictions that we had to accommodate for. The fine-tuning process locally was estimated around 19 hours. Therefore, we leveraged the McGill GPU and tmux to create an environment for our model to run more effectively and efficiently. In total, the fine-tuning process was reduced significantly.

However, we continued to run into disk space issues where resources were taking up significant space in correspondence to the existing data and model checkpoint files we had stored. Therefore, to accommodate for this we had to disable certain libraries such as $wandb$ to accommodate for the constrained computational resources. This allowed us to run 5 epochs under the training conditions we wanted.

For evaluating our proposed model during the fine-tuning process, we included metric generation for every epoch to see the trend over time. Our results section further details our findings in relation to our contributions.

5.2 Synthetic Data Generation Implementation

The current paper incorporates the following transformations to produce novel claims with CORRECT or INCORRECT labels attached: Paraphrasing, entity and number swapping, pronoun swapping, sentence negation, and noise injection. We additionally implement two additional transformation methods on our own: shuffling sentences and dropping words.

5.2.1 Shuffling Sentences

In this transformation, the sentences of the original source text are randomly swapped and placed back together into the proper text format. This allows us to generate a source text that invigorates more noise so that the model can effectively learn from various sentence orientations.

5.2.2 Dropping Words

This transformation incorporates randomly dropping words from the claim text based on an input probability (default value of 0.2) using the random module. For each word in the claim, a random number will be generated. If the random number is greater or equal to the dropout probability then the word will be included in the transformed claim text, otherwise it is not included. The collection of words is then joined back together into the proper claim text format. Similarly, this generates regularization and more robustness for the model when learning the sentence and word patterns.

To ensure proper comparison against the base model in terms of evaluating the performance, the implementations were not used in the training process, however, they add to the framework of our proposed model that can be used for future data generation. The proposed additions add to the complexity of synthetic data generation to further enhance the reliability of our proposed model.

6 Results

6.1 Fine-tuning Results

The fine-tuning process involved running the custom fine-tuning script produced by the original paper on the FactCC model using the training set extracted as a subset of the training data split in the PubMed dataset. At each epoch, the weighted balanced accuracy, F1-score, loss, and time taken were recorded and reported, as seen in Table 3.

Epoch	Balanced Accuracy (%)	F1-Score	Loss	Time Taken (s)
1	72.36	0.77	0.4485	444
2	79.26	0.8083	0.4485	447
3	77.98	0.8017	0.3869	448
4	80.20	0.825	0.4360	447
5	80.80	0.8267	0.5213	447

Table 3: The weighted balanced accuracy, F1-score, loss, and time taken for each epoch during the fine-tuning process of the FactCC model on the PubMed dataset

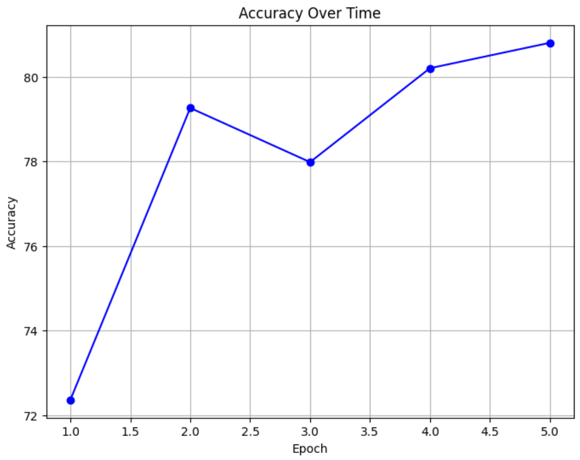

Figure 4: A plot showing the weighted balanced accuracy over each epoch for the FactCC model being fine-tuned on the PubMed dataset

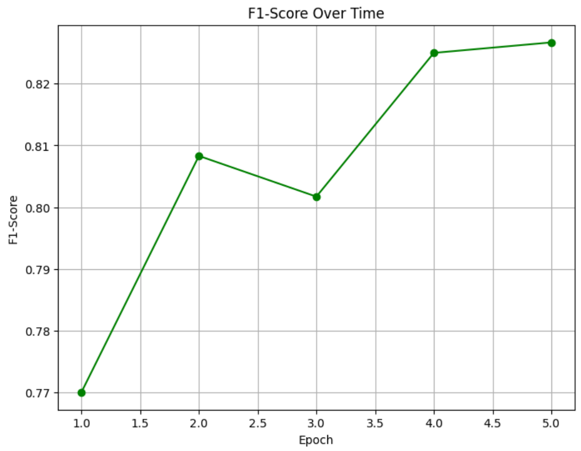

Figure 5: A plot showing the F1-score over each epoch for the FactCC model being fine-tuned on the PubMed dataset

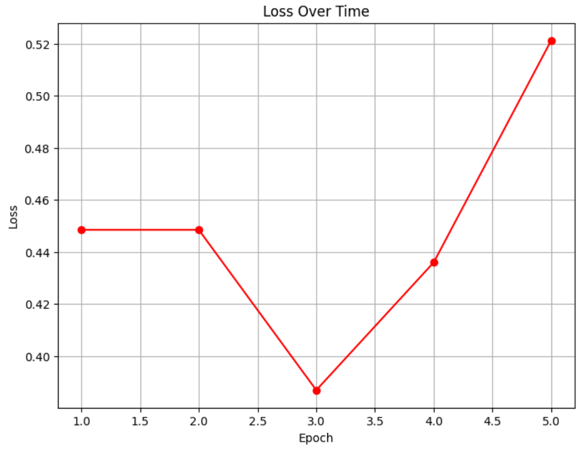

Figure 6: A plot showing the loss over each epoch for the FactCC model being fine-tuned on the PubMed dataset

Figure 4, Figure 5 and Figure 6 show the trendline of the weighted balanced accuracy, F1-score, and loss over 5 epochs during the fine-tuning process of the FactCC model on the PubMed dataset.

6.2 Evaluation Results

Model	Validation Results	Test Results
Baseline Model Paper (CNN/Dailymail)	Not Available	Balanced Accuracy: 74.15 F1-Score: 0.5106
Baseline Model (PubMed)	Balanced Accuracy: 72.00 F1-Score: 0.7766	Balanced Accuracy: 64.37 F1-Score: 0.7333
Custom FactCC Model (PubMed)	Balanced Accuracy: 79.30 F1-Score: 0.8072	Balanced Accuracy: 80.81 F1-Score: 0.8267

Table 4: A comparison of the weighted balanced accuracy and F1-scores between the validation and test sets of the baseline model as reported in the paper, the baseline model as run by ourselves, and our proposed custom model

In Table 4, the weighted balanced accuracy and F1-scores among the validation and test sets are reported with respect to three instances of the model: the baseline model as reported by the paper, the baseline model as reported and run by ourselves, and the proposed model following the fine-tuning process.

The baseline model reported in the paper, as trained on the CNN/Dailymail data, had a weighted balanced accuracy of 74.15 and an F1-score of 0.5106 on the manually annotated test set. In comparison, when evaluating the baseline model against the PubMed dataset, the test set gave a lower accuracy of 64.37 but a higher F1-score of 0.7333. Additionally, the baseline model has a weighted balanced accuracy of 72.00 and F1-score of 0.7766 when evaluated against the validation set of the PubMed dataset. Finally, our own custom FactCC model boasts a weighted balanced accuracy score of 79.30 and 80.81 for the validation and test set respectively, as well as an F1-score of 0.8072 for the validation set and an F1-score of 0.8267 for the test set. The improvement between the baseline before fine-tuning and after can be noted by the last two rows.

7 Discussions

7.1 Increasing Balanced Accuracy and F1-Scores during Fine-Tuning

Our fine-tuning process looked to analyze the weighted balance accuracy, F1-score, and loss over time as denoted in Table 3.

Despite only fine-tuning the model over 5 epochs, there is an obvious observed trend of increasing weighted balanced accuracy and F1-scores during the fine-tuning process. This shows that the FactCC model has strong capabilities for adapting and learning to the exposed PubMed dataset. The model shows improvement over time in its ability to effectively support the claims provided based on the source text within the biomedical domain.

Although there is a decrease in performance observed in one epoch for both accuracy and F1-score, the overall trend throughout the remaining epochs shows an upward trend. This indicates that although the model may have encountered challenges in one epoch, it was able to improve and recover on the subsequent ones. The stability and adaptability of the model can be highlighted by its ability to recover from challenges and setbacks through the fine-tuning process.

Overall, the increasing positive progression in the weighted balanced accuracy and F1-scores underscores our custom FactCC model's ability to verify factual consistency based on summarized biomedical text in the PubMed dataset. With even just 5 epochs, an increasing trend and boastful results can be seen, showing the significance of the fine-tuning process and the capabilities of this model when verifying claims.

7.2 Our Model Performance on Validation and Test Set

Table 4 presents the weighted balanced accuracy and F1-score of the baseline FactCC on the PubMed dataset and our custom fine-tuned FactCC model on the PubMed dataset.

In both the validation and test set, our custom fine-tuned FactCC model boasts a significant improvement in comparison to the baseline FactCC. In the validation set, our custom FactCC model has an accuracy of 79.30 % in comparison to the baseline FactCC accuracy of 72.00. Similarly, our custom FactCC model has an accuracy of 80.81 % in comparison to the baseline FactCC's accuracy of 64.37 %. This significant increase indicates that our fine-tuning process successfully enhanced the FactCC's model to detect conflicts between the source text and claims within the validation and test set. Therefore, this shows that our proposed custom FactCC model would be better suited for the biomedical domain when aiming to mitigate hallucinations produced by text summarization models.

For the F1-scores, our custom fine-tuned FactCC model also produced higher values than the baseline FactCC model. For our proposed model, we saw an F1-score of 0.8072 for the validation set and an F1-score of 0.8267 for the test set, while the baseline FactCC produced F1-scores of 0.7766 and 0.7333 for the validation and test set respectively. The larger F1-scores indicate that our custom model identifies true positives well while minimizing false positives and false negatives, providing us with more accurate information. With both higher precision and recall as indicated by our F1-scores, this shows that our custom fine-tuned FactCC is effective and trustworthy in identifying hallucinations produced by biomedical claim summaries.

7.3 Hallucinations Present In Biomedical Domain by BERT

One of the main inquiries of this project revolved around hallucinations generated by the BERT model during text summarization. As seen in the original paper (Kryscinski et al., 2020), the CNN/-Dailymail dataset that the model was trained on produced hallucinations that had to be mitigated by the fact-checking model. Consequently, this is also the case for the PubMed dataset where hallucinations are present when working with the uncased, base BERT model. This shows us that despite BERT being such a substantial language model, it is still prone to errors and requires other tools or technologies to combat some of the errors produced by it. Given the extensive use of the BERT model in our world today, specifically for text summarization, our research aims to highlight the hesitation that people should have when fully trusting the BERT model with summarizing texts. Instead, we propose that individuals use fact-checking models such as FactCC or their own custom models to combat any hallucinations that may appear so the result of utilizing text summarization is as accurate as possible for an ideal information transfer.

7.4 Limitations and Future Work

Given the scope and resource constraints of the project, we acknowledge that there are limitations that can be further evaluated in the future. While our results were positive and supported our hypothesis, we believe some future work can be pursued to increase the confidence of the results. Firstly, we were only able to achieve 5 epoch runs. While in our case, this was enough to show a positive trend from the baseline results, we acknowledge that the model can be further improved by more epochs and runs - an important feature to explore further. Additionally, we only chose a certain subset of each set of the split and therefore were not able to encompass all the examples of the PubMed dataset into our fine-tuning process. With more resources, we believe the use of more or all examples could only help further improve the quality of the proposed model. Lastly, while we did not incorporate our own synthetic data generation methods in our evaluation or fine-tuning process because we wanted an equal comparison to the baseline model, we believe future evaluation and fine-tuning using our methods will also help improve the diversity of the proposed model, as we incorporate unseen methods to generate data for the model to learn from.

8 Conclusion

With the surge in popularity of summarization systems like T5 and Bert, the increasing concerns of producing hallucinations or erroneous information is also relevant. Therefore, the implementation of these tools should be properly complemented by fact-checking tools, especially if the use of these technologies were in sensitive domains such as the biomedical domain where erroneous information transfer could be much more detrimental. Our study introduces a fine-tuned version of the FactCC model, tailored specifically for the PubMed dataset which is a commendable repository of biomedical literature. We leveraged the uncased, base BERT architecture and the fine-tuning process of the FactCC to produce a custom FactCC model that helps mitigate the presence of hallucinations that could appear in summarization systems. Our research mainly contributes by reporting the accuracy and fine-tuning protocol of the FactCC model on the PubMed dataset and additional synthetic data generation mechanisms in addition to the current implementations provided by the original paper.

Our study has shown that our fine-tuned custom FactCC model evaluates against the PubMed dataset significantly better than the baseline FactCC model on the same dataset in both weighted balanced accuracy and F1-score. This finding shows the importance and prevalence of integrating fact-checking mechanisms into text summarization processes, especially in the biomedical domain where text summarization can be very useful but important to navigate properly. We envision that our research paves the way for future explorations into leveraging text summarization for biomedical information with the help of confidently trained fact-checking mechanisms so all stakeholders involved can benefit within this domain.

9 Code

The code and results can be found here.

10 Contributions

Both members had equal contributions to the implementation, coding, and writing sections. Each component was brainstormed and agreed on by both members such that there was agreement on the direction of the project. Both members often worked together on the components simultaneously to ensure that there was consensus on all components of the project.

References

Muhammad Aurangzeb Ahmad, Ilker Yaramis, and Taposh Dutta Roy. 2023. Creating trustworthy llms: Dealing with hallucinations in healthcare ai. *arXiv*.

Amanuel Alambo, Tanvi Banerjee, Krishnaprasad Thirunarayan, and Michael Raymer. 2022. Entity-driven fact-aware abstractive summarization of biomedical literature. *arXiv*.

Hussam Alkaissi and Samy I McFarlane. 2023. Artificial hallucinations in chatgpt: Implications in scientific writing. *Cureus*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Meng Cao, Yue Dong, and Jackie Chi Kit Cheung. 2022a. Hallucinated but factual: Fact-guided language hallucination. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2985–2996, Online. Association for Computational Linguistics.

Meng Cao, Yue Dong, Jingyi He, and Jackie Chi Kit Cheung. 2022b. Learning with rejection for abstractive text summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9768–9780, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: fact-aware neural abstractive summarization. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

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

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.

Akhilesh Gupta, Nesime Tatbul, Ryan Marcus, Shengtian Zhou, Insup Lee, and Justin Gottschlich. 2021. Class-weighted evaluation metrics for imbalanced data classification.

Minghao Hu, Haoyu Wang, Bowen Tan, Wentao Wang, Zhengdong Lu, and Xiaoyong Du. 2020. Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1881–1892, Online. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, page 228–231, USA. Association for Computational Linguistics.

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

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 21, pages 10211–10222.

Thomas Scialom, Paul-Alexis Dray, Patrick Gallinari, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, and Alex Wang. 2021. Questeval: Summarization asks for fact-based evaluation. In *Conference on Empirical Methods in Natural Language Processing*.

Yu Sun, Shuohuan Cheng, Zhi Gan, Jiafeng Zhang, Jingjing Liu, Bing Liu, Chenghua Liu, Zheng-Jun Zha, and Hongyu Lin. 2020. Ernie 3.0: Large-scale knowledge enhanced pre-trained language model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9263–9280, Online. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2020. Bert rediscovers the classical NLP pipeline. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4597–4612, Online. Association for Computational Linguistics.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Wenpeng Yin, Dragomir Radev, and Caiming Xiong. 2021. DocNLI: A large-scale dataset for document-level natural language inference. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4913–4922, Online. Association for Computational Linguistics.