Addressing Factual Error in Abstractive Dialogue Summarization via Span Identification and Correction

Faiz Ghifari Haznitrama **Young-Jun Lee** **Ho-Jin Choi**

School of Computing, KAIST
{haznitrama, yj2961, hojinc}@kaist.ac.kr

Abstract

Abstractive dialogue summarization presents unique challenges due to the dynamic nature of conversations, involving multiple speakers, role changes, language variations, and informalities. Despite recent advancements in this field, summaries generated by existing methods often suffer from factual errors. To address this issue, post-processing correction has emerged as a promising approach that offers practicality and can be combined with other techniques. However, existing correction models still exhibit limitations, including false corrections that transform clean summaries into incorrect ones. We propose a simple and straightforward framework for correction with the main idea to separate the identification and use its results as guidance for better correction. Initially, the framework determines whether a summary contains factual errors and proceeds to identify the wrong part. This identified segment then serves as guidance for the correction. Our evaluation results demonstrated the effectiveness of our identifier and corrector model in terms of detecting incorrect summaries and performing corrections while highlighting its flexibility. Furthermore, the factuality human evaluation further emphasizes the ability of our approach to achieve accurate correction while preventing false correction.

1 Introduction

Abstractive dialogue summarization (Carletta et al., 2005) tackles the challenges posed by distilling essential information from multi-party conversations. Dialogue summarization exhibits distinctive characteristics compared to traditional article/document summarization, involving the dynamic nature of conversation such as multiple speakers, role changes, language variations, and informalities. Transformer-based (Vaswani et al., 2017) models such as BART (Lewis et al., 2020), T5 (Raffel et al., 2020), and Pegasus (Zhang et al., 2020) have made significant advancements (Chen and Yang, 2020; Malykh et al., 2020; Chen and Yang, 2021; Zhu et al., 2021), leveraging pre-training and fine-tuning. However, despite their successes, they tend to generate summaries with factual errors (Cao et al., 2018; Maynez et al., 2020; Kryściński et al., 2020; Lee et al., 2021; Zhong et al., 2021), which is a piece of incorrect information according to the dialogue contexts.

Researchers have aimed to address factual errors in abstractive dialogue summarization through various approaches, such as incorporating discourse-specific information (Zhu et al., 2020; Liu et al., 2021; Liu and Chen, 2021), applying constraints and filter generated candidates (Zhao et al., 2020; Mao et al., 2020), and modifying pre-training or fine-tuning objectives (Wan and Bansal, 2022; Cao and Wang, 2021; Tang et al., 2022). These efforts have shown promise in reducing factual errors but still face challenges in fully resolving the problem. In addition, post-processing correction, including training generative language models for correction on corrupted data (Cao et al., 2020; Lee et al., 2021) and treating correction as a question-answering task (Dong et al., 2020), have emerged as a promising approach. However, existing correction models face challenges in falsely correcting clean summaries and providing insufficient corrections, highlighting the need for improved techniques. Additionally, the application of post-processing correction methods in dialogue summarization is relatively unexplored, creating opportunities for advancements in this field.

This paper proposes a simple framework for post-processing correction in abstractive dialogue summarization. The main idea is to separate the identification and correction tasks, with a primary focus on utilizing the identified incorrect words as valuable guidance to improve correction. We explore three distinctive approaches for the identifier model: token classification, joint training with binary classification, and generative language mod-

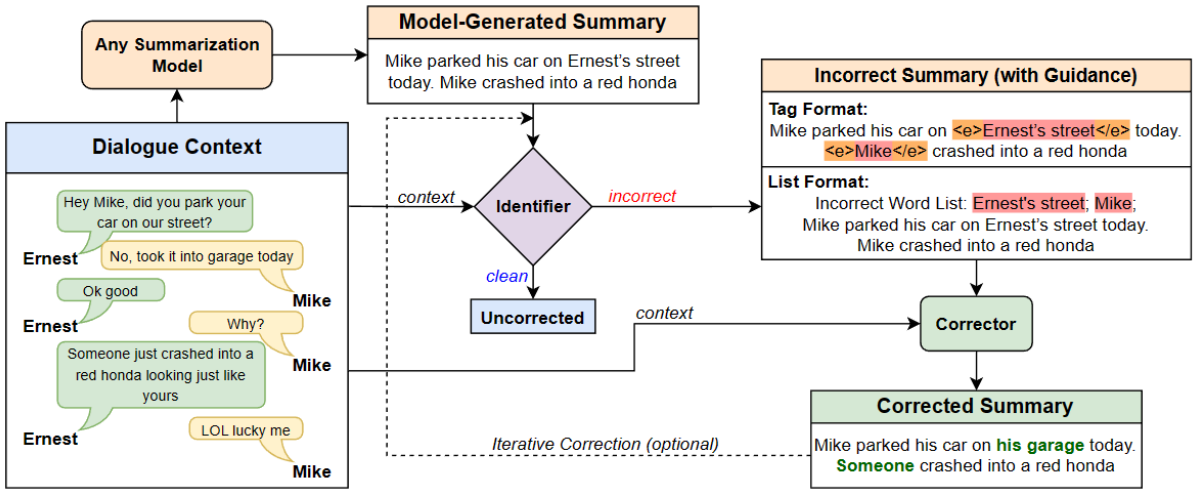


Figure 1: High-level illustration of our proposed framework, consisting of identifier and corrector model with an option for iterative correction.

eling for precise error identification. Additionally, the corrector model is designed to incorporate different guidance formats, including tagging incorrect words or providing an incorrect word list, to optimize the correction process. Through training these models, our research aims to contribute to the reduction of factual errors and the improvement of faithfulness in dialogue summaries.

Our contributions are as follows:

1. We propose a framework for dialogue summary correction with the idea of separating the identification and correction tasks and also utilizing the span identification as guidance to improve correction performance.

2. We explore different approaches to identify factual errors in summaries and highlight their importance in preventing false corrections.

3. We demonstrate the effectiveness of our proposed framework in improving corrections and reducing factual errors through empirical evaluations. We also showcase its practicality even when combined with baseline or existing methods.

2 Related Works

Post-processing correction is a flexible approach applied to generated summaries to address factual errors and semantic inconsistencies. Cao et al. (2020) introduced a post-processing correction method using BART (Lewis et al., 2020) to fix factual errors in corrupted summaries. Dong et al. (2020) proposed the Span-fact model, which leveraged Question Answering (QA) knowledge to correct multi-fact errors in summarization. They employed BERT (Devlin et al., 2019) as the basis for two models: span-fact and auto-regressive fact. Lee et al. (2021) addressed speaker incorrectness in dialogue summaries by training a BART-based speaker-focused corrector model. Their approach involved predicting the correction type and applying the necessary correction to the draft summary. FactCCX, a variant of FactCC (Kryściński et al., 2020) to assess the factual consistency of a summary, employs span selection heads to highlight supporting spans in the source text and identify error spans. This highlights the potential for leveraging these identified spans to improve the correction and enhancement of summaries.

Researchers who work on addressing factual errors use various techniques to generate corrupted summaries to train their models. These include text transformations (Kryściński et al., 2020), pronoun and entity swapping (Cao et al., 2020), error corruptions related to entities and numeric values (Zhao et al., 2020), and speaker-based manipulations (Lee et al., 2021). More complex corruption techniques like noun and verb swapping, number masking, and sentence deletion have also been employed (Tang et al., 2022). Furthermore, incorporating generative language models and techniques like source-conditioned regeneration and selecting the least probable generated sequence has been explored (Cao and Wang, 2021).

Correction and revision share similarities as they both aim to enhance the quality of a text. Deliterater (Kim et al., 2022) is a notable work in the field

Table 1: Example of transformed summary based on each corruption type. Red text highlights the corrupted words.

Reference	Andy'll go to the Georgian restaurant in Kazimierz on Saturday at 6 pm, and he'll pick her up on the way to the place.
Corruption Type	**Transformed Summary**
Speaker (Replace)	Lisa'll go to the Georgian restaurant in Kazimierz on Saturday at 6 pm, and he'll pick her up on the way to the place.
Speaker (Insert)	Andy and Robert'll go to the Georgian restaurant in Kazimierz on Saturday at 6 pm, and he'll pick her up on the way to the place.
Entity	Andy'll go to the Georgian restaurant in New York on Monday at 6 pm, and he'll pick her up on the way to the place.
Pronoun	Andy'll go to the Georgian restaurant in Kazimierz on Saturday at 6 pm, and she'll pick him up on the way to the place.
Verb	Andy'll pay to the Georgian restaurant in Kazimierz on Saturday at 6 pm, and he'll date her up on the way to the place.
Noun	Andy'll go to the Georgian way in Kazimierz on Saturday at 6 pm, and he'll pick her up on the restaurant to the place.
Number	Andy'll go to the Georgian restaurant in Kazimierz on Saturday at 4 pm, and he'll pick her up on the way to the place.

of text revision, employing a three-stage process: delineate, edit, and iterate. The system first detects editable spans, classifying tokens into revision intentions, then revising them based on their respective revision intentions. The process is repeated in the iterate stage until no editable spans are detected or the maximum revision depth is reached.

3 Proposed Framework

We propose a simple framework inspired by text revision (Kim et al., 2022) for post-processing correction to address summaries with potential factual errors, as illustrated in Figure 1. The main idea behind this framework is to separate the identification and correction tasks, leveraging the identification results as guidance to enhance the correction process. By specifically identifying the factual errors in a summary, we can focus our correction efforts on those specific areas, improving the accuracy and effectiveness of the correction process.

This framework includes three steps: Decide, Identify, and Correct. The Decide step determines if a summary has factual errors, while the Identify step pinpoints the specific incorrect content. The Correct step focuses on correcting the identified errors while maintaining the summary's meaning. The framework utilizes an identifier model for the Decide and Identify steps and a corrector model for the Correction step. Additionally, an iterative correction option allows for re-checking and further correction if needed. Training data is generated by intentionally corrupting reference summaries to create incorrect summaries.

3.1 Corrupted Summary Generation

Following previous works (Kryściński et al., 2020; Lee et al., 2021; Cao et al., 2020; Cao and Wang, 2021; Tang et al., 2022; Zhao et al., 2020), we employ various techniques to generate corrupted summaries to train our models. We focus on token-level corruption, targeting entities, pronouns, numbers, verbs, and nouns. For speaker corruption, we utilize insertion and replacement methods, omitting deletion unlike Lee et al. (2021). Replacement words/phrases are carefully selected from the dialogue source or reference summary, also we maintain case and pronoun groups for replaced pronouns. Numbers are replaced with the same type of number to maintain grammatical accuracy. We use spaCy (Honnibal et al., 2020) library to do the corruption. Detailed illustrations of the corruption can be found in Table 1.

3.2 Identifier Model

In our proposed framework, the identifier model plays a crucial role in deciding and identifying incorrect parts in a summary. The framework offers practicality and flexibility by allowing various models as identifier models, as long as they fulfill the objectives. We explore three options for the identi-

fier model: token classification, joint training with binary classification, and generative language modeling. In the token classification approach, each token is predicted to determine whether it is factually incorrect, employing the BIO tagging scheme with class labels for entities, pronouns, verbs, nouns, and numbers. The joint training approach combines binary classification and token classification tasks, where the model predicts if the summary is clean or incorrect, and if incorrect, identifies the specific error words. The generative model directly identifies incorrect words by generating summaries with tagged errors or providing a list of incorrect words.

3.3 Corrector Model

The corrector model is responsible for fixing the incorrect part of the draft summary identified by the identifier model. It utilizes the information of identified incorrect words to guide the correction process while preserving the overall meaning. Two guidance formats can be employed: tag and list. In the tag format, identified words are enclosed within tags based on token classification or generated text from the identifier model. This format provides explicit guidance by modifying the summary with tags. In contrast, the list format compiles identified words into a word list, which is used as additional input for the corrector model. The list format offers implicit guidance without directly marking the identified words, allowing flexibility to address corrections beyond the identified words. This approach is valuable when the identifier model may not capture all errors accurately, enabling the correction of words not included in the list.

4 Experimental Setup

4.1 Datasets

We use two datasets in our experiments: SAMSum (Gliwa et al., 2019) and DialogSum (Chen et al., 2021). The SAMSum dataset consists of 16,369 messenger-like conversations while DialogSum consists of 13,460 real-life conversations. Based on these two datasets, we create corrupted versions of them by utilizing the corruption techniques outlined earlier. The process involved three parameters: x controlled the ratio of corrupted data, y determined the probability of a corruption type being applied, and α represented the ratio of corrupted tokens within each sample for a specific corruption type.

Table 2: Corrupted SAMSum and DialogSum datasets statistics. E, V, P, N, and Q refer to entity, verb, pronoun, noun, and quantity (number) respectively. Speaker is considered an entity. The Test set is the results of human annotation on 300 samples of generated corrupted data.

Split	Clean	Corr.	Token Label				
			E	V	P	N	Q
SAMSum							
Train	7593	6928	10965	2702	6266	2591	173
Val	425	380	628	160	353	130	15
Test	419	208	333	82	205	77	8
DialogSum							
Train	6111	6030	23711	2656	3810	2800	68
Val	245	245	946	98	134	106	2
Test	268	195	716	80	100	69	1

Based on observations from previous studies (Lee et al., 2021; Tang et al., 2022; Cao et al., 2020), we prioritized corrupting speakers, entities, pronouns, and numbers while placing less emphasis on corrupting nouns and verbs. To generate the corrupted dataset, we used the following configurations: x = 0.5 (half of the data corrupted). For speakers, entities, pronouns, and numbers, y = 1.0, and α = 0.5. For nouns and verbs, y = 0.3, and α = 0.3. We excluded data exceeding a combined dialogue and summary length of 512 tokens to accommodate model limitations. Additionally, samples that remained unchanged after corruption were labeled as clean data. For detailed statistics regarding the generated corrupted dataset, please refer to Table 2.

Furthermore, we incorporate human annotation for the corrupted test set. We hired six annotators who were tasked with evaluating the quality of the generated corrupted summaries. A good corrupted summary was expected to contradict the dialogue while maintaining sound semantics, grammar, and syntax. We sampled 300 generated corrupted data samples from each of the SAMSum and DialogSum corrupted test sets. Each of these samples was evaluated by three different annotators, and the majority vote determined the result. Only the corrupted data that is evaluated as good will be used.

4.2 Models & Baseline

We used bert-large-uncased (Devlin et al., 2019) for the token classification and joint model. For the joint model, we use the joint implementation from JointBERT (Chen et al., 2019). The token classification model was trained for 5 epochs, while

Table 3: Identification experiment result. Bold represents the best score for each metric, and underline represents the second best.

Model	SAMSum		DialogSum	
	Token F_1	Acc.	Token F_1	Acc.
FEC	-	87.35	-	56.72
Our Identifier				
Token	<u>84.58</u>	**96.88**	81.06	<u>94.99</u>
Joint	83.17	95.09	<u>81.20</u>	93.85
Gen. (Tag)	81.89	<u>95.82</u>	72.51	**95.06**
Gen. (List)	**85.56**	90.45	**83.04**	79.94

the joint model was trained for 10 epochs, with both using a learning rate of 2e-5, weight decay of 0.01, dropout rate of 0.1, and a batch size of 8. The generative identifier and corrector model utilized bart-large (Lewis et al., 2020). We trained each model for 10 epochs, with a learning rate of 2e-5, batch size of 4, and a beam width of 6 for generation.

For the baseline, we compare our model with the **Factual Error Corrector (FEC)** (Cao et al., 2020) model. To ensure a fair comparison, the FEC model was trained with the same configuration as our corrector model on both clean and corrupted data. We also compare the corrected summary with the draft summary generated by **vanilla BART** (Lewis et al., 2020) model to assess the impact of correction in our framework in terms of factuality. All models used the HuggingFace (Wolf et al., 2019) implementation and were trained on a single Nvidia A100 GPU with 40GB of memory.

5 Experiments

5.1 Identification Experiment

In the identification experiment, we assess the model's ability to determine whether a summary contains factual errors and identify the specific incorrect parts. We evaluate two aspects: summary identification (binary classification) and incorrect span identification (token classification). We utilize balanced accuracy as the primary metric for the summary identification and the F1-score of the predicted tokens for incorrect span identification. For the FEC model, a summary is predicted to contain factual errors if the model makes any changes to the summary, as done in the paper (Cao et al., 2020).

Table 3 presents the identification evaluation results. All of our identifier models outperformed the FEC model in terms of determining whether a summary contains factually incorrect content. This result aligned with the data from Cao et al. (2020) that separating the identification task will result in better identification performance. However, in terms of identifying incorrect spans, our models can only achieve approximately 80% of the token F1-score, which indicates a suboptimal performance and might lead to less accurate guidance for correction in the next step. The Generative-List model as the best model in identifying incorrect spans, falls short in the summary identification, which indicates the tendency of the model to much more easily predict a summary to contain factual errors.

Table 4: Correction experiment result. Excluding the ideal condition, bold represents the best score for each metric, and underline represents the second best. CR. and R-L refer to Correction Ratio and ROUGE-L respectively.

Model		SAMSum		DialogSum	
Identifier	Corrector	CR.	R-L	CR.	R-L
FEC		<u>80.80</u>	**96.30**	<u>70.71</u>	**93.50**
Ours (Identifier + Corrector)					
Token	Tag	73.58	94.54	66.33	91.73
	List	77.11	95.43	68.35	92.56
Joint	Tag	75.12	94.60	65.99	92.09
	List	77.73	95.38	67.34	92.54
Generative	Tag	74.19	92.50	69.36	91.45
	List	**84.64**	<u>95.85</u>	**72.39**	<u>92.92</u>
Ideal Guidance (Theoretical Proof)					
Ideal	Tag	89.40	98.04	84.18	94.98
	List	88.63	97.48	79.46	94.10

5.2 Correction Experiment

In the correction experiment, we evaluate the model's performance in fixing incorrect words and spans, utilizing the corrupted data from the test set. We introduce a simple metric called the **Correction Ratio**. This metric calculates the ratio of correctly corrected corrupted tokens compared to all corrupted tokens, functioning as a recall metric specifically for correction. The correction ratio is calculated by simply dividing the number of correctly corrected corrupted tokens by the total number of corrupted tokens. The ability to compute the correction ratio is enabled by storing the corresponding correct tokens for each corrupted token during the corrupted dataset generation. We also use **ROUGE-L** (Lin, 2004) as it is considered to align with human factuality evaluation (Koh et al.,

Table 5: Evaluation result on the entire test set. Bold represents the best score for each metric, and underline represents the second best.

Model		SAMSum			DialogSum		
Identifier	Corrector	Acc.	Cor. Ratio	IDCR	Acc.	Cor. Ratio	IDCR
FEC		87.35	<u>80.80</u>	84.00	56.72	<u>70.71</u>	63.71
Our Identifier + Baseline Corrector							
Token	FEC	**96.89**	79.11	<u>87.99</u>	94.99	69.02	82.01
Joint		95.80	78.65	87.22	93.85	68.69	81.27
Generative		96.41	80.18	**88.30**	<u>95.69</u>	70.37	**83.03**
Ours (Identifier + Corrector)							
Token	Tag	<u>96.64</u>	73.58	85.11	95.55	66.33	80.94
	List	96.52	77.11	86.82	**96.69**	68.35	<u>82.52</u>
Joint	Tag	95.92	75.12	85.52	93.78	65.99	79.88
	List	96.40	77.73	87.07	93.89	67.34	80.62
Generative	Tag	95.82	74.19	85.00	95.06	69.36	82.21
	List	90.45	**84.64**	87.55	79.94	**72.39**	76.17

2022).

Table 4 provides the correction experiment's automatic evaluation results. In an ideal condition, the corrector model outperforms other models in terms of correction ratio. This outcome validates our theory that providing proper guidance leads to significant improvements in correcting summaries. However, when using the actual identifier model, the correction performance declines, although the Generative-List model still outperforms the baseline in correction ratio with slightly lower ROUGE-L. The lower performance of other configurations can be attributed to the wrong prediction of incorrect error spans, resulting in incorrect guides. The higher correction ratio of the Generative-List model also corresponds to the highest error identification in Table 3, as explained in the previous section.

5.3 Overall Performance

In this experiment, we assess the model's performance using the entire test dataset. We utilize a weighted metric, **IDCR** (Identification and Correction Ratio), to measure the combined performance of identification and correction. IDCR is calculated as the weighted sum of the identification balanced accuracy and correction ratio, with both having the same weight.

Table 5 shows the overall performance of each model. There we can see that our proposed identifier with corrector models performs better than the FEC model. Most of our models are particularly effective at accurately identifying clean summaries and preventing them from being wrongly corrected. This came from the separation between the identification and correction tasks in our proposed framework. The Generative-List model here came out as the best overall model for the SAMSum dataset, while the Token CLS-List model is much better in DialogSum because of the lower balanced accuracy from the Generative-List model in DialogSum. Our models provide different options that can be suited to different cases.

Table 6: Percentage of head-to-head human evaluation result for correction between our models compared to the FEC model. Krippendorff's α=0.81.

Model	SAMSum			DialogSum		
	Win	Lose	Tie	Win	Lose	Tie
vs Gen-List	**54**	26	20	**66**	12	22
vs Token-List	28	32	**40**	**52**	20	28
vs Joint-Tag	28	**36**	**36**	30	28	**42**

The FEC model, although having a high correction ratio, suffers from lower summary identification accuracy as its identification mechanism is combined with the correction. Specifically, in the DialogSum dataset where speaker names are censored with `#Person<Num>#`, which affected the model's tendency to predict most summaries to contain factual errors because lots of the appearance of symbol #. Such things did not happen in the SAMSum dataset which has its named person as it is.

One interesting part is we can combine our iden-

tifier model with the FEC model as the corrector. By doing so, we capitalize on the advantages offered by each model, thereby improving the overall performance. This also proves that our framework is flexible, allowing for the integration of various models that follow the framework's guidelines. However, utilizing the FEC model as a corrector means the identified incorrect span will not be used as the FEC model did not use any kind of guidance.

5.4 Human Evaluation

5.4.1 Correction

We conducted a human evaluation to further assess the correction performance. We sampled 50 corrected summaries from each dataset and compared three of our models head-to-head with the FEC model. Human evaluators were asked to determine which corrected summary was better in terms of correction and factuality aligned with the dialogue. Each comparison was evaluated by 3 different evaluators, and we took the majority answer as the result. We also compute Krippendorff's α-coefficient (Hayes and Krippendorff, 2007) to measure the inter-rater reliability.

Results in Table 6 prove that the Generative-List model consistently wins against the FEC model, as this model exhibits a higher correction ratio although slightly lower ROUGE-L. The Token-List and Joint-Tag models demonstrate comparable performance in most cases for the SAMSum dataset, while also producing better-corrected summaries in the case of DialogSum. These findings further emphasize the significance of providing accurate factual error words as guidance for correction. However, it should be noted that the effectiveness of the correction depends on the performance of the identifier model that provides the guide. The correction comparison sample used in this evaluation can be found in Figure 5 in the appendix section A.3.

5.4.2 Factuality

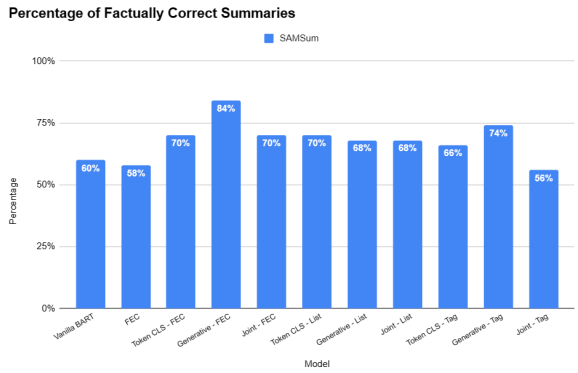
We focus on human evaluation to assess the factuality of summaries in comparison to the draft summaries without any correction, because automated metrics are unable to capture subtle differences from correction (automatic factuality evaluation result can be found in the appendix section A.2). In this evaluation, we presented the dialogue, one draft summary generated by Vanilla BART, and ten summaries (corrected or not) produced by ten different model configurations, including the FEC model and models with our framework. We sampled 50 instances from each dataset in which at least one model performed a correction. We ask human evaluators to check each one of the summaries provided whether is considered factually correct or not. Each sample was evaluated by 3 different evaluators, and we take the majority vote as the final decision. We also compute Krippendorff's α-coefficient (Hayes and Krippendorff, 2007) to measure the inter-rater reliability.

Figure 2 provides the result of the factuality human evaluation. In the case of the SAMSum dataset, most models from our proposed framework demonstrated the ability to generate more factually correct summaries compared to the draft. This highlights the optimal decision-making capability of our framework in determining when to perform corrections and when to retain the original draft. Additionally, the FEC model tended to make more corrections on the draft, resulting in a higher incidence of false corrections and slightly fewer factually correct summaries overall. These findings align with our earlier experiment results, which indicated that the FEC model suffers from lower identification accuracy, leading to false corrections. For the DialogSum dataset, most of our models also result in the same or more factually correct summaries, while the FEC model also still has a lower percentage because of false corrections. Sample summaries used in this factuality evaluation can be found in Figure 6 in the appendix section A.4.

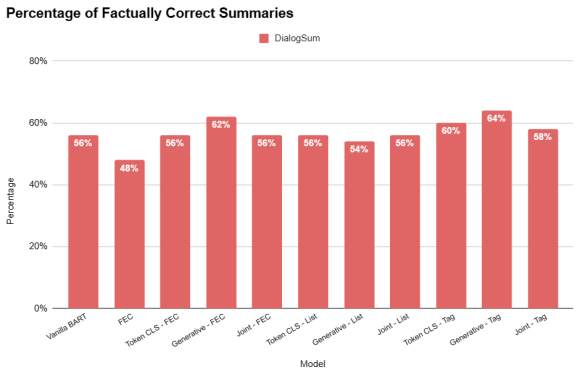
6 Discussion

We highlight some important findings and considerations regarding the performance and characteristics of the proposed correction framework. First, there exists a trade-off between correction ratio and identification accuracy, as models with higher correction ratios tend to exhibit lower identification accuracy. Finding the right balance between easily correcting summaries and accurately identifying incorrect parts is crucial for optimizing the framework's overall performance. In regards to identification, generative models show superior performance compared to other models, showing that generation currently is the way to go for most tasks.

From the correction experiment, we can see that the list format for guidance is more robust compared to the tag format. While the tag format performs better in ideal conditions, the list format proves preferable especially when the identifier



(a) SAMSum. Krippendorff's α=0.54



(b) DialogSum. Krippendorff's α=0.39

Figure 2: Percentage of factually correct summaries based according to human evaluators.

model's guidance may be incomplete or inaccurate. Also, we acknowledge the limitations of relying solely on automated metrics to measure the factuality of summaries, which we provide in the appendix section A.2. Automated metrics may not capture subtle differences that determine the faithfulness of a summary, emphasizing the need for human evaluation.

7 Conclusion

In this paper, we proposed a simple framework for addressing factual errors in abstractive dialogue summarization with the main idea of separating the identification task and utilizing guidance for correction. Our identifier models achieved superior identification accuracy, confirming the strength of our approach to detecting factual errors. We also showed that providing proper guidance can improve correction performance. Overall, our framework outperformed the baseline in balancing identification and correction tasks and proved flexible. The human evaluation further validated the effectiveness of our models and framework, with improved correction and higher factuality according to human evaluators.

8 Future Work

Future work can focus on improving the identification accuracy of the identifier model, particularly for the token and joint models, to match the effectiveness of the generative model in providing guidance for correction. Additionally, exploring correction at higher levels such as sentences or entire summaries presents an interesting direction. This would involve incorporating information about factual errors in these higher-level units and developing new corruption techniques and guidance methods to train the corrector model accordingly. These advancements would contribute to improving the overall performance and capabilities of this framework.

9 Limitations

Limitations of our work are observed when utilizing the actual identifier model, leading to a decline in the correction performance. This is attributed to the reliance of the corrector model on the guidance provided by the identifier model, where the token and joint identifier models achieved an approximate F1-score of 80%, implying that some incorrect tokens/spans may go unidentified. Consequently, the correction performance is affected by errors or incompleteness in the guidance, emphasizing the need for accurate and comprehensive guidance.

Additionally, the correction in our framework operates at the token/span level, making it challenging to directly measure the impact of correction using automated metrics. Even slight changes in token/span level correction may not yield significant differences in automated metric scores. Furthermore, there are cases where token/span level correction is insufficient as the entire summary or sentence may be flawed, highlighting the necessity for higher-level correction methods that involve reconstructing sentences or summaries.

Ethical Considerations

We acknowledge the limitations of our correction models and the possibility of false corrections. Our models should not be considered as providing absolute truth, and it is important to exercise criti-

cal analysis and verify information from reliable sources. We also made sure all human annotators and evaluators are participate voluntarily, were fairly compensated, and given informed consent for their participation.

References

Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. 2020. Factual error correction for abstractive summarization models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6251–6258.

Shuyang Cao and Lu Wang. 2021. Cliff: Contrastive learning for improving faithfulness and factuality in abstractive summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6633–6649.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Jiaao Chen and Diyi Yang. 2020. Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4106–4118.

Jiaao Chen and Diyi Yang. 2021. Structure-aware abstractive conversation summarization via discourse and action graphs. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1380–1391.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. Dialogsum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074.

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.

Yue Dong, Shuohang Wang, Zhe Gan, Yu Cheng, Jackie Chi Kit Cheung, and Jingjing Liu. 2020. Multi-fact correction in abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9320–9331.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79.

Andrew F Hayes and Klaus Krippendorff. 2007. Answering the call for a standard reliability measure for coding data. *Communication methods and measures*, 1(1):77–89.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, et al. 2020. spacy: Industrial-strength natural language processing in python.

Zae Myung Kim, Wanyu Du, Vipul Raheja, Dhruv Kumar, and Dongyeop Kang. 2022. Improving iterative text revision by learning where to edit from other revision tasks. *arXiv preprint arXiv:2212.01350*.

Huan Yee Koh, Jiaxin Ju, He Zhang, Ming Liu, and Shirui Pan. 2022. How far are we from robust long abstractive summarization? *arXiv preprint arXiv:2210.16732*.

Wojciech Kryściński, 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.

Philippe Laban, Tobias Schnabel, Paul Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Dongyub Lee, Jungwoo Lim, Taesun Whang, Chanhee Lee, Seungwoo Cho, Mingun Park, and Heui-Seok Lim. 2021. Capturing speaker incorrectness: Speaker-focused post-correction for abstractive dialogue summarization. In *Proceedings of the Third Workshop on New Frontiers in Summarization*, pages 65–73.

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.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Zhengyuan Liu and Nancy Chen. 2021. Controllable neural dialogue summarization with personal named entity planning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 92–106.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2021. Coreference-aware dialogue summarization. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 509–519.

Valentin Malykh, Konstantin Chernis, Ekaterina Artemova, and Irina Piontkovskaya. 2020. Sumtitles: a summarization dataset with low extractiveness. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5718–5730.

Yuning Mao, Xiang Ren, Heng Ji, and Jiawei Han. 2020. Constrained abstractive summarization: Preserving factual consistency with constrained generation. *arXiv preprint arXiv:2010.12723*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Xiangru Tang, Arjun Nair, Borui Wang, Bingyao Wang, Jai Desai, Aaron Wade, Haoran Li, Asli Celikyilmaz, Yashar Mehdad, and Dragomir Radev. 2022. Confit: Toward faithful dialogue summarization with linguistically-informed contrastive fine-tuning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5657–5668.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

David Wan and Mohit Bansal. 2022. Factpegasus: Factuality-aware pre-training and fine-tuning for abstractive summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1010–1028.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zheng Zhao, Shay B Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, et al. 2021. Qmsum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. Mediasum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934.

Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang. 2020. A hierarchical network for abstractive meeting summarization with cross-domain pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 194–203.

A Appendix

A.1 Ablation Studies

We conducted some ablation studies to provide an in-depth analysis of some parts of our proposed framework.

A.1.1 Corruption Type Label

We explore the impact of different corruption type labels. By default, the corrupted tokens were labeled according to their specific corruption types. However, we also investigated the performance when combining all corruption types into a single label, thus simplifying the task.

The ablation studies in Table 7 demonstrate that combining all corruption types into one label leads to a decrease in identification performance across all models. The models appear to be more effective at predicting incorrect words when they are labeled based on their specific corruption types. This outcome is expected since the separation of corruption types makes the task more similar to other common tasks, such as part-of-speech (POS) tagging. The corruption type also acts as an implicit hint for the identifier model, resulting in better token classification performance. However, few models appear to have better accuracy although still have lower token classification performance.

We also investigated the impact of different configuration corruption labels on the correction performance, as shown in Table 8. We found that using combined corruption labels generally resulted in lower performance for the correction task in most models. The lower correction performance observed when using combined corruption labels can be attributed to the lower identification performance. The correction model heavily relies on the guide provided by the identifier model. When the identification performance is compromised due to the combined corruption labels, it impacts the correction model's ability to make accurate corrections. However, models with the list guidance format showed a different pattern. These models convert the output of the identifier model into a word list, which focuses on the correct identification of words or spans rather than their specific token positions. As a result, the performance of models with list guidance format appears to be less affected by corruption labels.

Table 7: Ablation study with corruption labels on identification experiment. Bold represents the best score for each metric, and underline represents the second best.

Model	**Label**	**SAMSum**		**DialogSum**	
		Token F_1	Acc.	Token F_1	Acc.
Token	Sep.	**84.58**	**96.88**	<u>81.06</u>	94.99
	Comb.	82.28	<u>95.92</u>	78.54	**96.76**
Joint	Sep.	<u>83.17</u>	95.09	**81.20**	93.85
	Comb.	81.44	<u>95.92</u>	75.04	93.73
Gen. (Tag)	Sep.	-	95.82	-	<u>95.06</u>
	Comb.		95.34		93.91

Table 8: Ablation study with corruption labels on correction experiment. Bold represents the best score for each metric, and underline represents the second best.

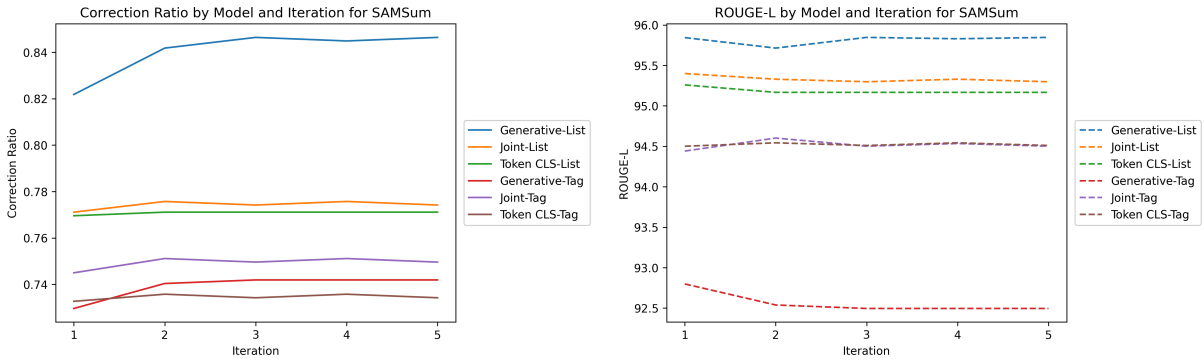
Model		**Label**	**SAMSum**	**DialogSum**
Identifier	Corrector		Cor. Ratio	Cor. Ratio
Token	Tag	Sep.	73.27	65.66
		Comb.	71.43	65.99
	List	Sep.	76.96	67.68
		Comb.	<u>77.11</u>	**68.35**
Joint	Tag	Sep.	74.50	65.32
		Comb.	72.66	63.64
	List	Sep.	<u>77.11</u>	66.67
		Comb.	**77.57**	67.34
Gen.	Tag	Sep.	72.96	<u>68.01</u>
		Comb.	66.82	62.63

A.1.2 Iterative Correction

As mentioned, our framework offers the option to apply the correction process iteratively, allowing for multiple rounds of checking and correcting the identified errors in the summary. We conducted an ablation study to investigate the effects of iterative correction on model performance as shown in Figure 4 and Figure 3. In this study, we experiment with the number of iterations from 1 (no iteration) until 5, and the iterative process would continue until either the identifier model predicted the summary as clean or the maximum iteration was reached.

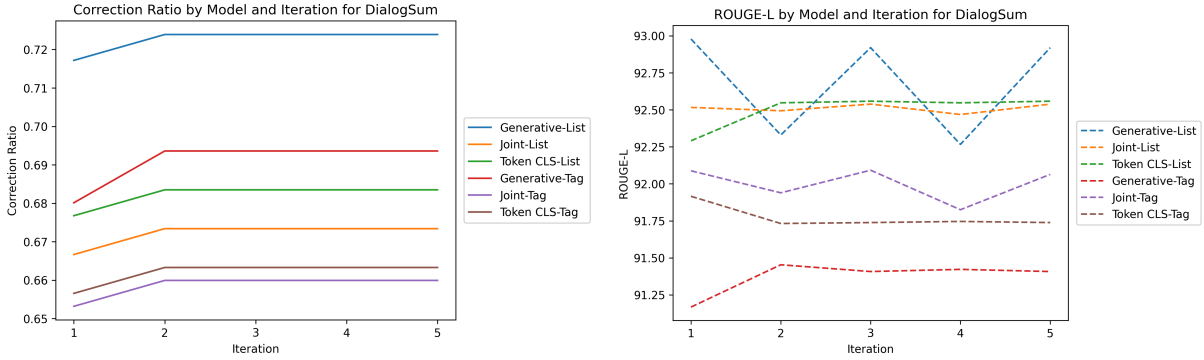
Overall, the results indicate a slight improvement in the correction ratio when applying iterative correction across all models. This suggests that running the framework iteratively provides the model with additional opportunities to review and correct errors that may have been missed in the previous iterations. The iterative nature of the process allows for a more thorough examination of the summary and can contribute to a higher correction ratio.

However, it is important to note that there is a decrease in the ROUGE-L score for some models



(a) Correction Ratio (b) ROUGE-L

Figure 3: Correction ratio and ROUGE-L by model and number of iterations for SAMSum dataset



(a) Correction Ratio (b) ROUGE-L

Figure 4: Correction ratio and ROUGE-L by model and number of iterations for DialogSum dataset

when iterative correction is applied. The decline in the ROUGE score suggests that when applying iterative correction, the corrector model tends to modify not only the identified corrupted tokens but also correct words. This phenomenon leads to a higher correction ratio but a lower ROUGE score since unnecessary changes may be introduced, affecting the alignment with the reference summary. In this case, we should treat the number of iterations differently for each model, because they have different behavior and optimal number of iterations to maximize both the correction ratio and ROUGE-L score. For example, the Generative-List model in the SAMSum dataset achieves optimal performance with 3 iterations, where the correction ratio is maximized and the ROUGE-L score comes back up after a decline in iteration 2.

A.2 Automatic Factuality Evaluation

The evaluation results presented in Tables 9 and 10 provide insights into the factuality evaluation using some automated metrics for the SAMSum and DialogSum datasets. **ROUGE** (Lin, 2004) evaluates n-gram overlaps between the generated and reference summaries. **BERTScore** (Zhang et al., 2019) which measures the similarity between the generated text and the reference using BERT. **BARTScore** (Yuan et al., 2021) which formulates the evaluation of generated text from BART. **SummaC** (Laban et al., 2022) a factuality metric that relies on NLI models by comparing the generated summary to the source.

Analyzing the results, it is noticeable that the scores across the various metrics do not exhibit substantial variations. The corrector models do not exhibit significantly higher scores compared to the vanilla BART model. This suggests that the corrections made by the corrector models are relatively minor, involving only a few words or subtle modifications. However, it is crucial to recognize that even small changes in the summaries can have a considerable impact on their factuality. Changes in names or pronouns, for instance, can significantly influence the faithfulness of the summaries. Unfortunately, the automated metrics might not fully capture these changes, resulting in similar scores for

Table 9: Factuality evaluation results using automated metric for SAMSum dataset. Bold represents the best score for each metric.

Model		ROUGE			BERT.S	BART.S	SummaC	
Identifier	Corrector	R-1	R-2	R-L	F_1	F_1	ZS	Conv
Vanilla BART		51.00	26.03	41.73	91.63	-2.77	1.02	34.52
Corrected Draft								
FEC		51.03	**26.22**	**41.80**	91.63	-2.77	1.05	**34.69**
Token CLS	FEC	51.04	26.11	41.73	**91.64**	-2.77	0.97	34.49
	List	**51.05**	26.04	41.72	91.63	-2.77	1.11	34.51
	Tag	51.03	26.10	41.73	91.63	-2.77	0.93	34.49
Joint	FEC	51.02	26.11	41.74	91.63	-2.77	1.00	34.52
	List	51.03	26.05	41.73	91.63	-2.77	1.11	34.54
	Tag	51.04	26.08	41.74	91.63	-2.77	0.84	34.47
Generative	FEC	50.99	26.11	41.73	91.63	-2.77	1.03	34.57
	List	50.97	25.94	41.64	91.62	-2.78	1.13	34.52
	Tag	50.99	26.10	41.72	91.63	-2.77	**1.17**	34.60

Table 10: Factuality evaluation results using automated metric for DialogSum dataset. Bold represents the best score for each metric.

Model		ROUGE			BERT.S	BART.S	SummaC	
Identifier	Corrector	R-1	R-2	R-L	F_1	F_1	ZS	Conv
Vanilla BART		43.95	17.96	**35.01**	91.18	-2.74	-49.79	27.39
Corrected Draft								
FEC		43.67	17.86	34.86	**91.19**	-2.74	-49.39	27.48
Token CLS	FEC	43.94	17.96	35.00	91.18	-2.74	-49.79	27.39
	List	43.94	17.98	34.99	91.18	-2.74	-49.77	27.39
	Tag	43.93	17.94	34.98	91.18	-2.74	-49.76	27.39
Joint	FEC	43.95	17.97	**35.01**	91.18	-2.74	-49.83	27.35
	List	**43.96**	**17.99**	**35.01**	91.18	-2.74	-49.78	27.39
	Tag	43.94	17.96	35.00	91.18	-2.74	-49.92	27.34
Generative	FEC	43.87	17.98	**35.01**	91.17	-2.75	-49.63	27.49
	List	43.62	17.82	34.79	91.17	-2.76	**-49.28**	27.56
	Tag	43.78	17.93	34.92	91.15	-2.75	-49.29	**27.79**

the corrector models and the vanilla BART model. These evaluations highlight the challenges of assessing factuality through automated metrics.

A.3 Correction Evaluation Sample

We provide some samples that were used for correction comparison human evaluation, including the evaluator's answer. The correction comparison samples are in Figure 5.

A.4 Factuality Evaluation Sample

We provide some samples that were used for factuality human evaluation. The summary samples can be found in Figure 6.

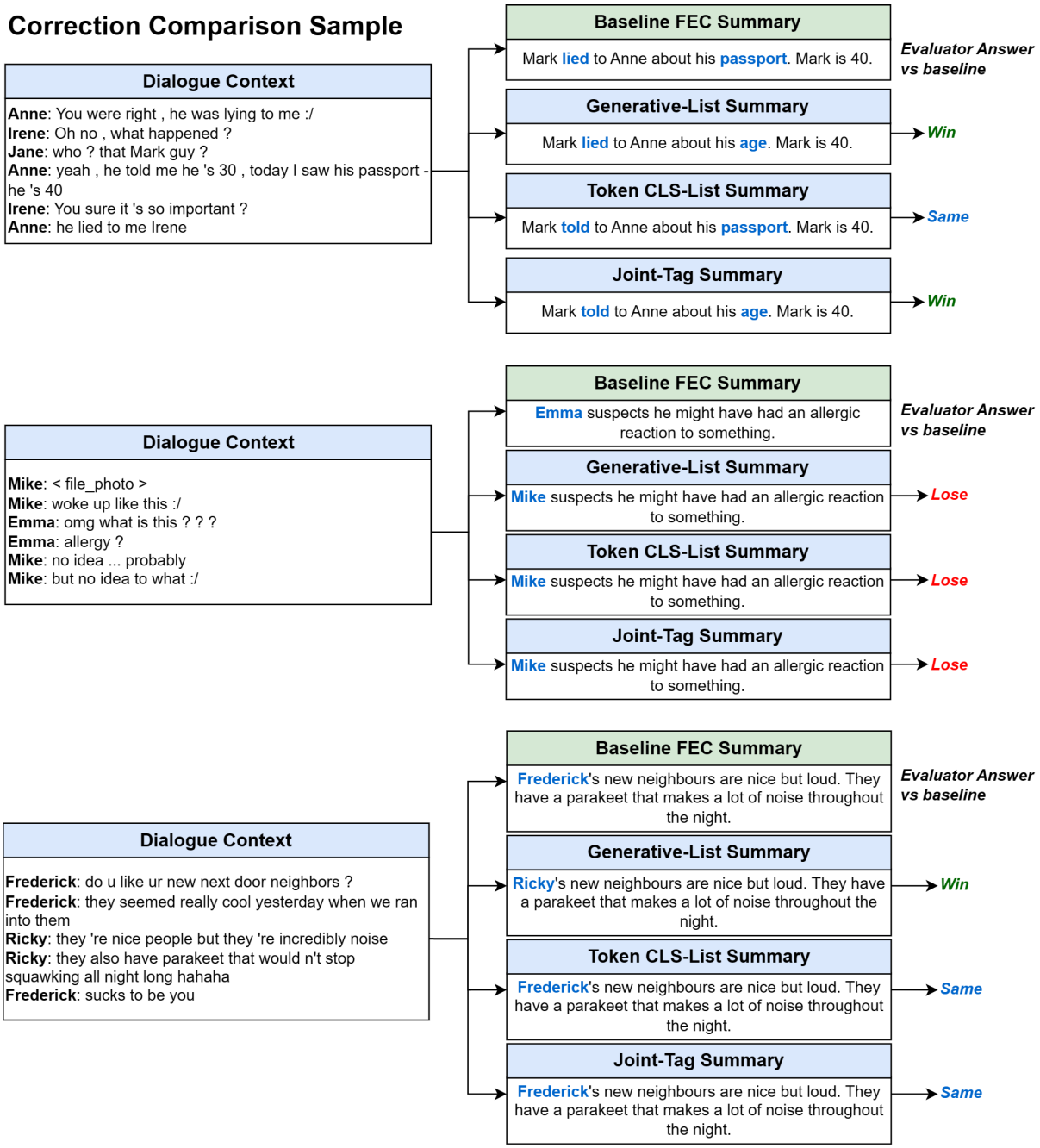


Figure 5: Sample of correction comparison between FEC, Generative-List, Token CLS-List, and Joint-Tag model. The evaluator's answer in comparison to the FEC model is provided on the right side. The blue text highlights different corrections performed by the models.

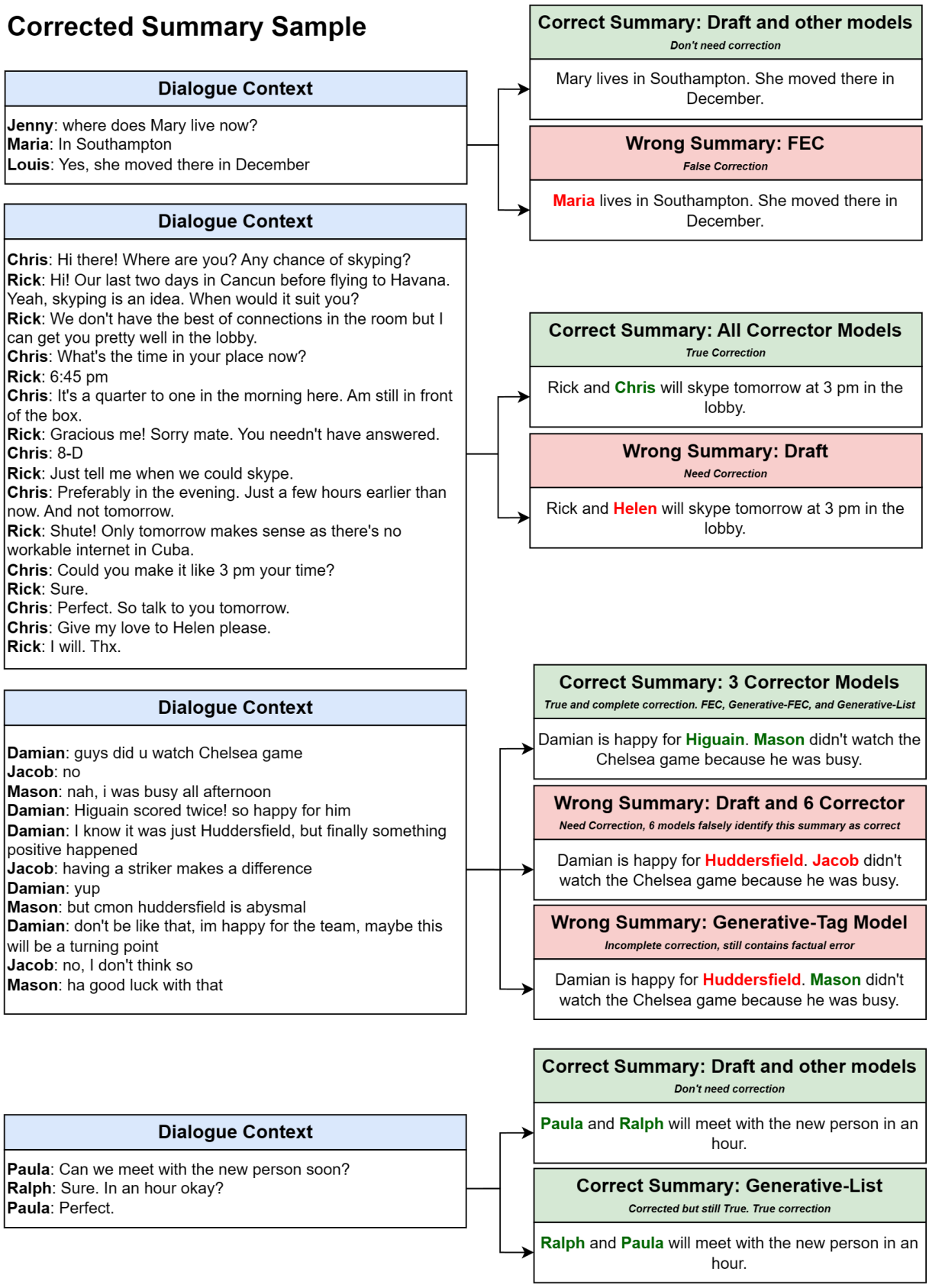


Figure 6: Sample of correction performed to the draft summary from BART. Red text highlights factually incorrect words, green text highlights words that are successfully corrected.