Contextualizing Air Traffic Management Conversations using Natural Language Understanding

Aida Sharif Rohani*
NASA Ames Research Center, Moffett Field, CA 94035, USA

David Nielsen
KBR, NASA Ames Research Center, Moffett Field, CA 94035, USA

James Della-Giustina
NASA Ames Research Center, Moffett Field, CA 94035, USA

Krishna Kalyanam
NASA Ames Research Center, Moffett Field, CA 94035, USA

I. Abstract

Efficient management of air traffic and mitigation of delays depend on extracting actionable information from unstructured data, such as dialogues from the Federal Aviation Administration's (FAA's) Air Traffic Control System Command Center (ATCSCC) telecons. This study presents a pipeline utilizing Natural Language Processing (NLP) methods for Intent Classification (IC) and Slot Filling (SF) to identify and extract Traffic Management Initiatives (TMIs) from aviation-specific dialogues. We leveraged DeBERTa, a pre-trained transformer model, and fine-tuned it to the nuances of the aviation domain. Despite challenges posed by annotation complexities, the IC model achieved promising results with a weighted average F1-score of 0.81. Our results are close to those of human annotators, which demonstrates the model's strong alignment with human-level performance. The SF model also showed strong performance, achieving a weighted F1-score of 0.97, which demonstrates its effectiveness in accurately predicting key slots. Our analysis revealed limitations in handling less frequent intents and slot labels due to data sparsity, motivating future efforts to adopt joint IC-SF modeling and data augmentation strategies. This research highlights the potential of domain-specific NLP to streamline decision-making in the aviation industry and improve the management of TMIs.

II. Introduction

The Federal Aviation Administration's (FAA) Air Traffic Control System Command Center (ATCSCC) plays a pivotal role in regulating air traffic, especially during adverse weather conditions or airport constraints that stress the U.S. National Airspace System (NAS). Experts from government agencies and the aviation industry manage traffic and balance capacity with demand through a collaborative decision-making process. Using daily teleconference calls (or telecons), various stakeholders including Air Traffic Control (ATC) centers, Terminal Radar Approach Control Facilities (TRACONs), Towers, and Aviation industry partners discuss flight planning under weather constraints, runway closures, and other potential issues[1]. One integral component of these conference calls is engaging in discussions regarding Traffic Management Initiatives (TMIs). TMIs are techniques used by Air Traffic Controllers (ATCos) to balance demand with capacity when conditions are not ideal, either at an airport or in a section of airspace. One example of a TMI is the Ground Delay Program (GDP), which involves delaying aircraft at their departure airport to reconcile demand with capacity at their arrival airport. Another example is Ground Stop (GS) which is a procedure that requires aircraft that meet specific criteria to remain on the ground at their origination airport. Following each telecon, ATC specialists compile and publish detailed ATCSCC advisories. The advisories include information regarding airspace restrictions, weather conditions, TMIs, and other relevant updates that affect flight operations and are crucial to air traffic users and other stakeholders.

A manual analysis of telecons is a time-consuming, tedious process and requires considerable resources. Therefore, the goal is to develop an automated system that accurately retrieves key or specific information from the recorded audio of telecons. Harnessing cutting-edge Natural Language Processing (NLP) techniques enables comprehension of these

*aida.sharifrohani@nasa.gov

audio files and the extraction of essential points from the conversation.

III. Related work

NLP techniques have been widely applied in the aviation domain for various tasks. This research focuses on extracting key information, particularly TMI-related sentences and attributes, from telecon transcripts. NLP applications in aviation have demonstrated effectiveness in various contexts, such as classifying and organizing safety-critical information. For instance, Tanguy et al. [2] used text classification to categorize safety reports based on the causes of incidents or accidents. They employed a supervised machine learning model (SML) to assign each report the appropriate label, showing how classification models can be used to manage and interpret complex aviation data. A fundamental issue with SML algorithms is their reliance on the availability of large, labeled datasets for efficient training. To address this issue, several other studies [3–5] avoided training a classification model from scratch and instead fine-tuned a pre-trained Large Language Model (LLM), such as RoBERTa [6], for their classification task. Recent advancements include Aviation-BERT [7], a domain-specific model pre-trained on accident and incident text narratives from the National Transportation Safety Board (NTSB) and Aviation Safety Reporting System (ASRS) databases, which has been shown to outperform general-purpose BERT in aviation text-mining tasks. Matthews et al., [8] used RoBERTa for sentiment analysis of aviation safety reports using ASRS dataset followed by clustering using the HDBSCAN [9] method to glean common corrective actions taken by pilots. In another study, Badrinath et al. [10], used rule-based techniques and a Named Entity Recognition (NER) model to extract call-signs and runway identifiers from ATC-Pilot communication transcripts. Wang et al., [11] proposed a method to transform complex unstructured ATCo's commands into simple structured ones. They employed Automatic Speech recognition (ASR) to transcribe ATC audio commands followed by the application of NLP techniques such as semantic labeling and NER to analyze the transcript and eventually obtain the structured instruction.

Despite significant progress in applying NLP techniques to aviation tasks, several challenges remain. Existing approaches, such as those by Matthews et al., [8] and Badrinath et al. [10], focus primarily on predefined structured outputs or specific tasks, such as extracting call signs or clustering sentiment-based corrective actions. While these methods demonstrate the utility of NLP for aviation safety, they often rely on either manual rule-based systems or lack robustness in handling domain-specific variations, such as those seen in telecon dialogues. Our work builds on these methods by addressing a critical gap in the aviation NLP domain: extracting key information directly from conversational dialogues, which presents unique challenges compared to structured data like incident reports or call signs. The unstructured nature of telecon dialogues, requires a novel approach to both labeling and structuring the data. This work directly extracts sentence-level intents as a crucial first step. Following this, the focus shifts to extracting TMI-related attributes from these TMI-related sentences, as these attributes are crucial for effective air traffic control decision-making. Natural language understanding (NLU) techniques are employed to close the gap between unstructured communication data and specific actionable insights. This hierarchical approach addresses issues such as complex relationships within unstructured text and offers a more robust solution compared to static rule-based systems or single-task models.

IV. Proposed approach

NLU focuses on understanding and interpreting human language in a meaningful way. *Intent Classification* (IC) and *Slot Filling* (SF) — with SF referring specifically to the NLP domain and not any FAA process — are two key techniques that, when applied in Air Traffic Management (ATM), can significantly aid the decision-making processes. IC involves the categorization of user statements to find the intention behind them. By automatically categorizing relevant text into predefined intents, users can focus on the most critical tasks without having to read through long telecon transcripts (or listen to the audio). SF is the task of extracting specific information, such as dates, times, and quantities from the user's input. It complements IC by identifying key details within the text that are relevant to the determined intent.

This study focuses on identifying the intents behind telecon phrases, followed by extracting detailed information specifically related to TMIs. For instance, in a sentence describing a planned TMI, the IC model first determines the sentence's intent as TMI-related. The SF model then extracts key TMI attributes, such as type, scope, status, and underlying reasons (e.g., poor visibility or runway closure). By combining IC and SF, our approach efficiently distills actionable insights from extensive telecon transcripts. Figure 1 illustrates a sample sentence, where the IC model predicts the intent, followed by the SF model extracting the relevant attributes.

Figure 2 provides an overview of our process, starting with the transcription of telecons and progressing to training

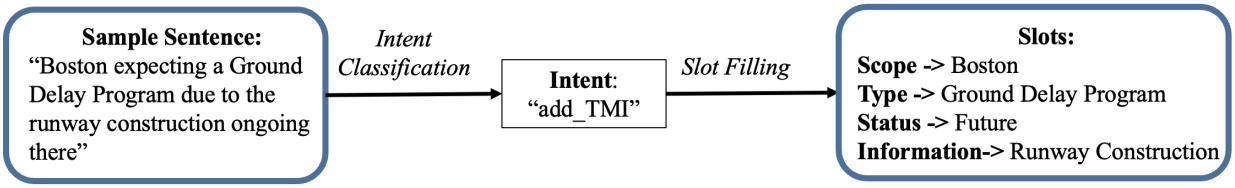


Fig. 1 A sample teleconference sentence is first categorized by IC, and subsequently, the slots are extracted through SF.

both IC and SF models, illustrating the relationship between the two. Starting with 25 ATCSCC telecons, comprising a total of 5 hours of audio, we first divided the audio files into 1-minute segments. These segments were then transcribed into text using our in-house, fine-tuned Whisper model [12]. After transcription, Subject Matter Experts (SMEs) manually reviewed and refined the unformatted text. The compiled text then underwent preprocessing steps, including Inverse Text Normalization (ITN) and sentence splitting by punctuation marks (e.g., question marks and periods). ITN was used to convert unformatted text into a more readable form for end-users: for instance, converting words to numerical representations (e.g., "thirty-eight" to "38"), standardizing temporal expressions (e.g., "seventeen hundred zulu" to "1700Z"), and formatting TMI names according to FAA standards (e.g., "seattle g d ps" to "Seattle GDPs"). For further details on the ITN step, refer to the work by Guo et al. [13].

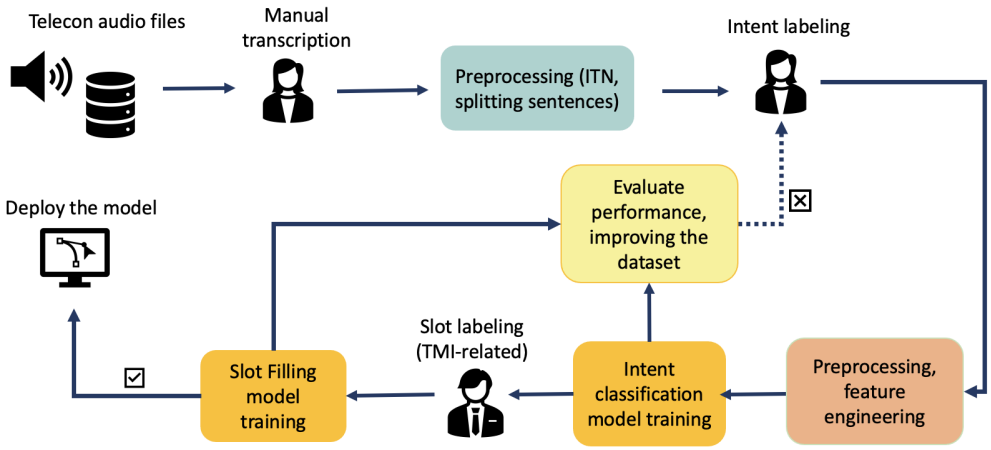


Fig. 2 Overview of the IC and SF tasks.

V. Datasets

After processing, the individual sentences were organized into an IC dataset of 4,300 real-world examples from telecon transcriptions. Using the Prodigy software [14] as an interface, Subject Matter Experts (SMEs) annotated each sentence with one of the nine labels defined in Table 1. The labels were categorized into general aviation issues, including flight arrival rate changes (*AAR_change*), TMI-related topics (*add_TMI*), airport constraints (*add_airport_constraint*), weather constraints (*add_weather_constraint*), removal of TMIs or constraints (*remove_item*), and changes in flight routes (route). Additionally, some labels reflected more natural language scenarios, such as requests for updates from stakeholders by ATC (*getting_updates*) or reports of positive situations at stakeholder bases (*give_positive_report*). Sentences that did not convey any specific meaning or intention were labeled as *none*.

However, due to the complexity of the telecons and the conversational nature of the sentences, creating the dataset was particularly challenging. It was often difficult to determine which label to assign to a given sentence, as multiple intents could occur simultaneously or the distinctions between labels were subtle, requiring careful judgment. To assess how challenging this task was and evaluate annotator agreement, a subset of the dataset (150 sentences) was tagged by four different SMEs based on the definitions provided for each label in Table 1. To find the agreement between

annotators, Fleiss' Kappa [15] score was calculated. Fleiss' Kappa is a statistical measure for inter-rater reliability, assessing the degree of agreement among multiple raters, where a score of 0 indicates no agreement and 1 indicates perfect agreement. For the dataset subset, the Fleiss' Kappa score was 0.35, indicating 'fair agreement' among the annotators. Insights from this analysis guided multiple rounds of dataset refinement to ensure each sentence was assigned a single, most-suitable label. This iterative process was crucial for creating a consistent and reliable dataset for downstream tasks.

Table 1 Intent definitions and examples.

Intent	Definition	Example
AAR_change	A change in the arrival rate at an airport	"Looks like they're operating at the 34 rate with the trips"
add_TMI	An indication that there is a need to modify/add a TMI (or a strategic plan) or a potential for a future TMI/strategic plan	"We are managing with some mile in trail there"
add_airport_constraint	A constraint was added due to a condition at an airport, it can be a possible constraint	"We have some overage here, low ceilings and runway construction"
add_weather_constraint	A constraint was added due to weather conditions, it can be a possible constraint	"Sparse coverage throughout Florida"
getting_updates	Asking stakeholders for updates	"New York center anything you'd like to add?"
give_positive_report	An indication that the conditions are desirable at a center/airport	"Yeah, the TAF showing actually clearing, yeah, between 13 and 14Z"
none	General discussions that do not have a specific intent	"Thanks for that Potomac"
remove_item	An indication that a TMI, airport constraint, weather constraint, or route will be removed or canceled early, prior to its scheduled end time	"San Francisco showing clearing sometime around the 1530"
route	Any conversation regarding routes	"Newark and Kennedy wind routes are out"

The SF dataset was constructed by selecting all sentences labeled as add_TMI from the IC dataset, focusing on extracting relevant TMI attributes. This selection was made because TMIs are considered one of the most critical aspects of the telecons and extracting the TMI-related attributes can facilitate and expedite decision making for the command center traffic specialists. The SF dataset was comprised of 550 sentences out of the total 4,300 sentences from the telecons.

Each slot (or label) has its own definition and role in describing TMIs comprehensively. However, similar to the IC dataset creation, the conversational format of telecons presents a challenge, as not all slots are stated within a single sentence. Another significant challenge during data annotation was determining the appropriate label for each token, as there were often multiple ways to assign slot labels to entities—or even decide whether an entity should be assigned a slot at all. To address this, we conducted multiple rounds of tagging with different domain experts, but certain instances still lacked consensus on the correct slot, highlighting the inherent difficulty of labeling conversational data. The final dataset was created only after several group discussions among the experts to reach a consensus on each sentence, ensuring consistency and accuracy in the annotations. Our goal was to align the selected TMI attributes with the FAA's Flow Information Exchange Model (FLXM). The selected TMI attributes include the type of TMI (Type), the scope of the TMI (Scope), the current status of the TMI (Status), with examples such as *current*, *future*, or *possible*. Additionally, the attributes capture any information explaining why the TMI was implemented (Information), arrival rate information within the TMI context (*Rate*), the timing of the upcoming TMI (*Time*), and relevant details about the TMI's impact on arrivals or departures (*Statistics*). The complete list of TMI slots for the SF task is provided in Table 2. For each attribute, an example sentence is provided in the third column, with the corresponding attribute highlighted in bold for clarity.

For SF dataset tagging, the SMEs used the Prodigy software interface to label relevant slots, applying the IOB

(Inside-Outside-Beginning) tagging scheme at the token level to mark each slot's boundaries within the text. Each character is tagged as:

- B- (Beginning): Marks the start of a token.
- I- (Inside): Marks characters continuing within the same slot.
- O (Outside): Marks characters that are not part of any slot.

Figure 3 shows an example of how a sentence from the dataset is tagged using the IOB scheme.

Sentence:	We'll certainly be talking with Seattle in probably about an hour or so about the GDP probabilities.
IOB slots:	O O O O O B-scope O B-status B-time I-time I-time O O O O B-type B-status O
Intent:	"add_TMI"

Fig. 3 IOB tagging of a sentence from the dataset.

Table 2 TMI-attributes definitions and examples.

TMI attribute	Definition	Example
Type	The type of the TMI being discussed	"Seattle expecting a **GDP** due to the runway construction ongoing there"
Scope	The scope of where the TMI applies	"**Seattle** expecting a GDP due to the runway construction ongoing there"
Status	Indicates the status of the TMI: current, future, or possible	"Seattle **expecting** a GDP due to the runway construction ongoing there"
Information	Represents causal factors to the TMI	"Seattle expecting a GDP due to the **runway construction** ongoing there"
Rate	Arrival rate information	"We do some around **28 30** airborne"
Time	The temporal information about a TMI	"I was told Toronto extended their GS until **1215Z** and it's for all aircraft"
Statistics	used for delay statistics and other TMI statistical details	"I was told Toronto extended their GS until 1215Z and **it's for all aircraft**"

VI. AI Model Description

For both IC and SF tasks, BERT [16], a pre-trained large language model developed by Google AI in 2018, was utilized. Trained on a vast corpus of text, BERT provides a solid foundation for a wide range of NLP tasks. However, fine-tuning is essential to adapt BERT to the unique terminology and phraseology of the aviation domain. Unlike training a model from scratch, which requires large amounts of labeled data, fine-tuning allows for adjusting BERT's parameters to better fit the specific dataset and tasks at hand.

BERT-based models, such as Distilled BERT, DeBERTa (both large and base variants), and uncased BERT, are optimized versions of the original BERT model, each designed to improve performance, reduce computational overhead, or handle specific nuances like casing in text, offering flexibility for various natural language processing tasks. DeBERTa Large was used for model training due to its enhanced performance over traditional BERT models. DeBERTa [17] improves upon BERT by incorporating a more efficient attention mechanism and better handling of word dependencies, which allows it to capture richer semantic information. Its larger model size provides greater capacity to understand complex language patterns, making it particularly well-suited for the specialized terminology and structure of the aviation domain. After training, the model's performance is evaluated by analyzing its predictions and identifying areas for improvement. The datasets are then iteratively refined and improved to enhance accuracy and achieve the desired performance level.

The following metrics were used to evaluate the model training:

- Precision is the ratio of correctly predicted positive observations to the total predicted positives. It is defined as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

where TP is the number of true positives, and FP is the number of false positives.

- Recall is the ratio of correctly predicted positive observations to all observations in the actual class. It is defined as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

where TP is the number of true positives, and FN is the number of false negatives.

- The F1-score is the harmonic mean of precision and recall, and it is calculated as:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Each of the IC and SF datasets were split into 80% for training and 20% for testing. After the split, the sentences were tokenized and further processed in preparation for the model training. The training datasets were subjected to 5-fold cross-validation, where it was partitioned into five subsets. During each fold, one subset was used for validation, and the remaining four subsets were used for training. This technique ensures that the model is evaluated on different portions of the training data, improving its generalization and reducing the risk of overfitting.

VII. Results

The distribution of intents in the IC dataset is shown in Figure 4. The dataset is imbalanced, with the majority of sentences labeled as either *none* or *getting_updates* indicating that they do not contain any useful information. Following those two majority classes, the most frequent intents are weather constraints (*add_weather_constraint*), TMI-related (*add_TMI*), and airport constraints (*add_airport_constraint*), respectively.

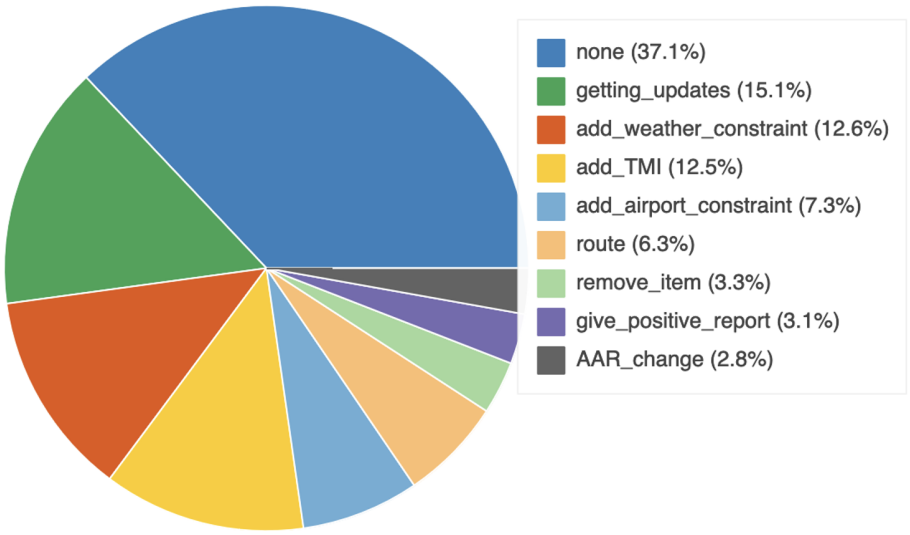


Fig. 4 IC dataset distribution.

DeBERTa Large was fine-tuned on the IC dataset, and the evaluation results are presented in Table 3. Varying performance is observed across different intents. For instance, the *AAR_change* intent has high precision (0.93) but a lower recall (0.74), suggesting that the model is accurate when predicting this intent but misses some instances. In contrast, *add_airport_constraint* and *give_positive_report* exhibit lower precision and recall (0.47 and 0.50, respectively), indicating that the model struggles to consistently identify these intents. On the other hand, *add_weather_constraint* and *getting_updates* show strong performance with high F1-scores (0.85 and 0.87), reflecting balanced precision and recall, and suggesting that the model accurately identifies these intents without significant errors.

Overall, the macro average F1-score of 0.73 reflects the model's average performance across all intents, while the weighted average F1-score of 0.81 demonstrates better performance on more frequent intents (such as *none*). The accuracy of 0.80 indicates that the model has a solid understanding of the dataset, though there is still room for improvement in handling less frequent intents. To compare the model's performance with the initial SME-tagged 150 sentences, all four labels for each sentence were evaluated. If at least one annotator selected the correct label, it was recorded in a separate column as a correct label, and the same classification report was generated. Annotator scores of 0.74 for F1-score and 0.83 for accuracy were obtained, showing close alignment with the model's classification scores, indicating consistency between the initial annotations and the model's predictions.

Table 3 Evaluation results for IC.

Intent	Precision	Recall	F1-score
AAR_change	0.93	0.74	0.82
add_TMI	0.80	0.81	0.80
add_airport_constraint	0.47	0.52	0.50
add_weather_constraint	0.88	0.83	0.85
getting_updates	0.89	0.84	0.87
give_positive_report	0.50	0.47	0.49
none	0.87	0.87	0.87
remove_item	0.42	0.67	0.51
route	0.85	0.83	0.84
accuracy	0.80	0.80	0.80
macro avg	0.73	0.73	0.73
weighted avg	0.81	0.80	0.81

To evaluate the IC model's performance in more detail, the confusion matrix shown in Figure 5 is examined to identify challenging predictions and instances where certain labels, such as *add_airport_constraint*, were inaccurately predicted. Analysis of the matrix reveals that the model frequently misclassified *add_airport_constraint* as *add_weather_constraint*, *add_TMI*, or *AAR_change*.

DeBERTa Large was fine-tuned on the SF dataset, and the overall weighted F1 score for the test set is 0.97. Instead of presenting multiple evaluation metrics such as F1-score, accuracy, and precision for the SF model training results, the confusion matrix is studied due to the complexity introduced by the IOB (Inside, Outside, Beginning) format. The confusion matrix in Figure 6 illustrates the performance of the SF model, showing the alignment between true and predicted labels across multiple attribute types. The diagonal entries represent correctly classified instances for each label, while off-diagonal entries indicate misclassifications. For example, the model shows relatively strong performance on key labels such as B-scope with 41 correct predictions and B-type with 37 correct predictions, suggesting reliable detection of these key categories. However, some misclassifications are evident, such as instances of B-information and I-information predicted as O (outside), as well as instances of I-status and B-status classified incorrectly across various other labels. Since the information and status slots are less critical and serve to add additional details rather than being essential for the core task, misclassifications in these categories are more forgivable. There is a notable class imbalance, as seen from the high count of O predictions (4003 instances), indicating the predominance of outside entities in IOB tagging. This demonstrates that the model is strongly capable of selecting the slots of interest and accurately dismissing irrelevant ones.

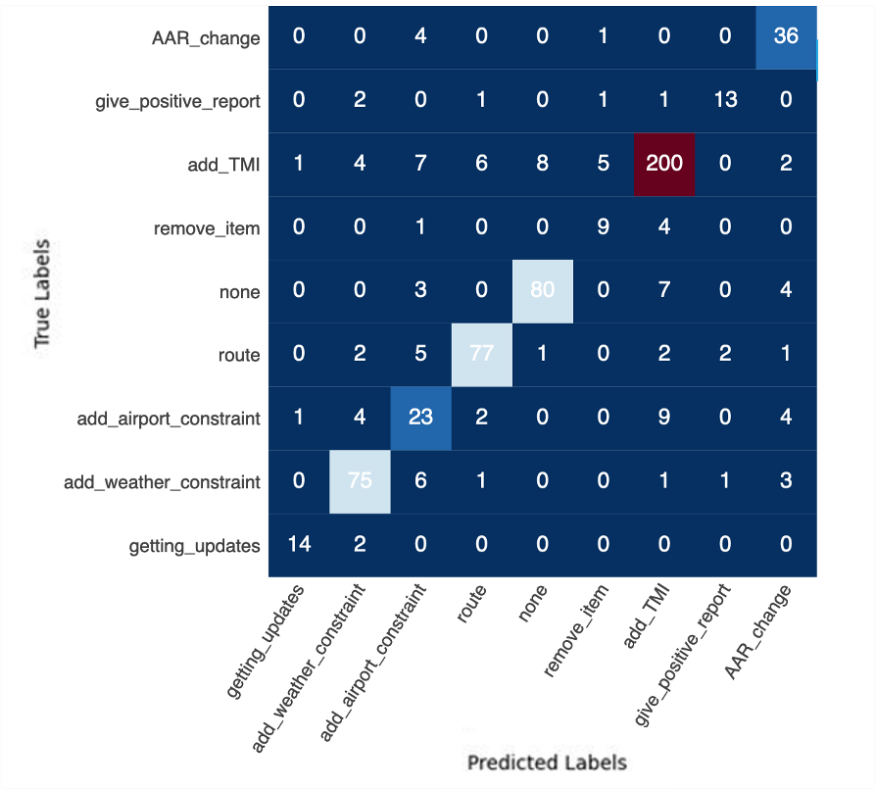
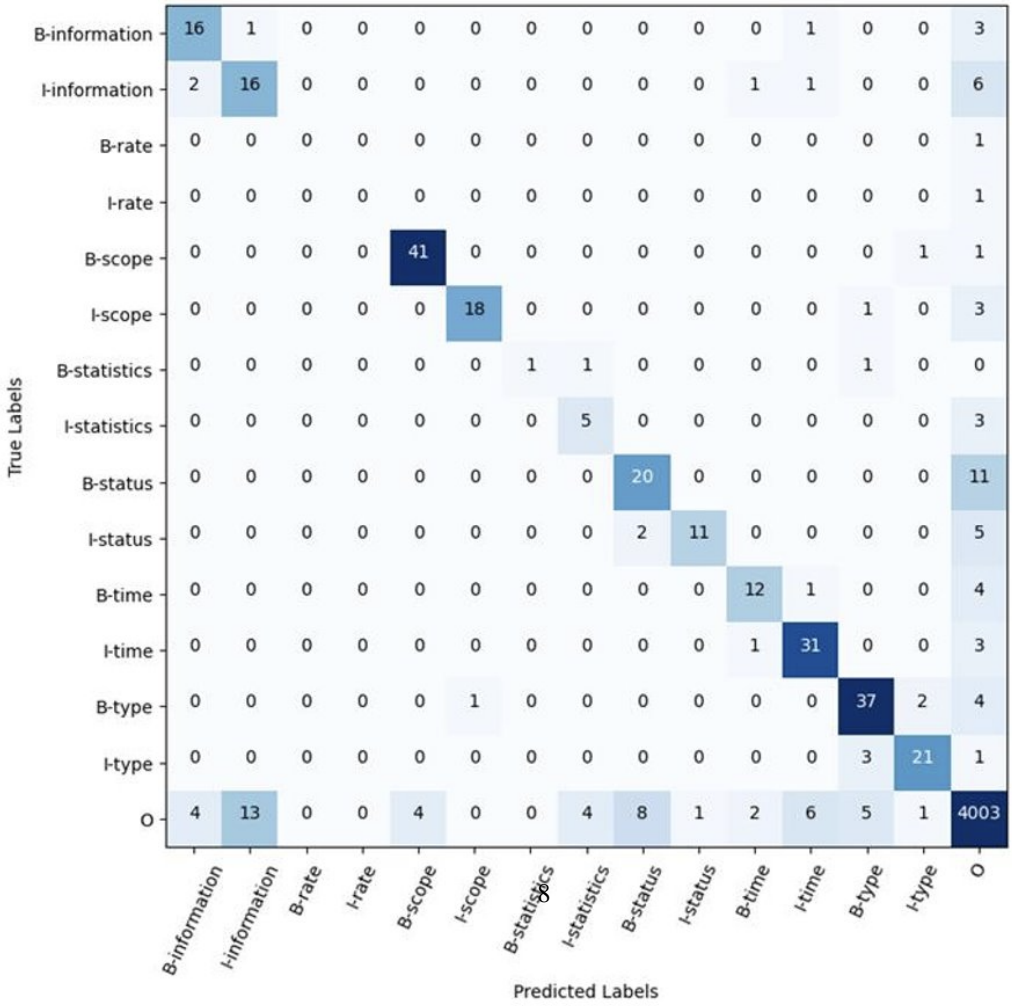


Fig. 5 Confusion matrix for IC.



VIII. Conclusions

In this study, we focused on extracting critical information, specifically TMIs, from ATCSCC's telecons in the aviation domain. To achieve this, aviation domain experts manually tagged two datasets: one for Intent Classification and one for Slot Filling. Tagging these datasets was particularly challenging due to the conversational nature of the sentences, which introduced complexity in annotation. In a small subset of the data, annotator agreement was only 0.35 for the IC dataset, which shows the difficulty in achieving consistent tagging. Therefore, it required several rounds of tagging and revisions to acquire the final datasets. A BERT-based model called DeBERTa was employed, which was fine-tuned for both the IC and SF models. The evaluation results demonstrate promising performance in classifying telecon sentences by their intent and accurately identifying key attributes related to TMIs within each sentence. When compared to the initial SME-tagged 150 sentences, the model's performance aligns closely with human annotators. To calculate the human annotator score, for each sentence, all four labels were evaluated at once, and if at least one annotator selected the correct label, it was marked as correct and stored in a separate column. Using this approach, the model performance was calculated and we achieved a weighted F1-score of 0.81, compared to 0.88 for human annotators, and an accuracy of 0.80, compared to 0.83 for the human annotators. This demonstrates that the model's predictions are very close to those of human annotators, indicating its strong capability in replicating human-level annotation.

For the SF model, a weighted average F1-score of 0.97 was obtained, demonstrating the model's strong capability in accurately selecting the relevant slots of interest while effectively dismissing outside slots (as indicated by the IOB tagging). This high score highlights the model's ability to focus on key attributes related to TMIs and shows its robustness in distinguishing between important slots and non-relevant information. Moving forward, our efforts will focus on developing a joint IC and SF model that can process both tasks simultaneously. Additionally, we aim to explore data augmentation and active-learning techniques to address the challenges posed by manual dataset tagging, which is time-consuming and resource-intensive. These advancements will help improve model performance and efficiency in real-world applications.

Acknowledgments

The authors acknowledge the invaluable support and feedback received from subject matter experts affiliated with the Federal Aviation Administration's (FAA) Office of NextGen (ANG) and Mr. Kari Gonter from SimLabs.

References

[1] "Federal Aviation Administration. Air Traffic Control System Command Center (ATCSCC)," `https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/systemops/nas_ops/atcscc`, 2024.

[2] Tanguy, L., Tulechki, N., Urieli, A., Hermann, E., and Raynal, C., "Natural language processing for aviation safety reports: From classification to interactive analysis," *Computers in Industry*, Vol. 78, 2016, pp. 80–96.

[3] Dong, T., Yang, Q., Ebadi, N., Luo, X. R., and Rad, P., "Identifying incident causal factors to improve aviation transportation safety: Proposing a deep learning approach," *Journal of Advanced Transportation*, 2021, pp. 1–15. https://doi.org/10.1155/2021/5540046.

[4] Klein, T., Lapasset, L., and Kierszbaum, S., "Transformer-based model on aviation incident reports," *CORIA 2021*, 2021. URL https://hal.archives-ouvertes.fr/hal-03200916.

[5] Marev, K., and Georgiev, K., "Automated aviation occurrences categorization," *2019 International Conference on Military Technologies (ICMT)*, 2019, pp. 1–5. https://doi.org/10.1109/MILTECHS.2019.8870055.

[6] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V., "ROBERTa: A robustly optimized bert pre-training approach," *CoRR*, 2019. https://doi.org/10.48550/arXiv.1907.11692.

[7] Chandra, C., Jing, X., Bendarkar, M. V., Sawant, K., Elias, L. R., Kirby, M., and Mavris, D. N., "Aviation-BERT: A Preliminary Aviation-Specific Natural Language Model," *Proceedings of the Aerospace Systems Design Laboratory Conference*, 2023.

[8] Matthes, B., Barshi, I., and Feldman, J., "An Approach to Identifying Aspects of Positive Pilot Behavior within the Aviation Safety Reporting System," *42nd Digital Avionics Systems Conference (DASC)*, 2023.

[9] Campello, R. J. G. B., Moulavi, D., and Sander, J., "Density-Based Clustering Based on Hierarchical Density Estimates," *Advances in Knowledge Discovery and Data Mining*, edited by J. Pei, V. S. Tseng, L. Cao, H. Motoda, and G. Xu, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 160–172.

[10] Badrinath, S., and Balakrishnan, H., "Automatic speech recognition for air traffic control communications," *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2676, No. 1, 2022, pp. 798–810. https://doi.org/10.1177/03611981211036359.

[11] Wang, X., Wang, G., and Xu, Q.-C., "A new structural template design of control instruction for semantic analysis," *CICTP 2019*, 2019, pp. 2935–2945. https://doi.org/10.1061/9780784482292.254.

[12] OpenAI, "Whisper: Automatic Speech Recognition System," https://github.com/openai/whisper, 2022.

[13] Guo, K., Clarke, S. B., and Kalyanam, K. M., "Inverse Text Normalization of Air Traffic Control System Command Center Planning Telecon Transcriptions," *AIAA AVIATION Forum and Exposition*, 2024.

[14] AI, E., "Prodigy: Annotation Tool for Machine Learning," https://prodi.gy, 2024.

[15] Fleiss, J. L., "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, Vol. 76, No. 5, 1971, pp. 378–382.

[16] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Google AI Language*, 2018.

[17] He, P., Liu, X., Gao, J., and Chen, W., "DeBERTa: Decoding-enhanced BERT with Disentangled Attention," *CoRR*, Vol. abs/2006.03654, 2020. URL https://arxiv.org/abs/2006.03654.