Article

Not peer-reviewed version

Aviation-BERT-NER: Named Entity Recognition for Aviation Safety Reports

Chetan Chandra *, Yuga Ojima, Mayank V. Bendarkar, Dimitri N. Mavris

Posted Date: 9 October 2024

doi: 10.20944/preprints202410.0656.v1

Keywords: Natural Language Processing; Text Mining; Aviation Safety; NTSB; ASRS; Named Entity Recognition

Article

Aviation-BERT-NER: Named Entity Recognition for Aviation Safety Reports

Chetan Chandra [1,2,*], Yuga Ojima [1,2], Mayank V. Bendarkar [2,3] and Dimitri N. Mavris [2,4]

[1] Graduate Researcher
[2] Aerospace Systems Design Laboratory (ASDL), Georgia Institute of Technology, Atlanta, GA, 30332, U.S.A
[3] Research Engineer II
[4] Regents Professor and Director, ASDL
* Correspondence: cchandra7@gatech.edu

Abstract: This work introduces Aviation-BERT-NER, a Named Entity Recognition (NER) system tailored for aviation safety reports, building on the Aviation-BERT base model developed at the Georgia Institute of Technology's Aerospace Systems Design Laboratory. Unlike general NER models, this system integrates aviation domain-specific data, including aircraft types, manufacturers, quantities, and aviation terminology, to enhance its precision in identifying and classifying named entities critical for aviation safety analysis. A key innovation of Aviation-BERT-NER is its template-based approach to fine-tuning, which utilizes structured datasets to generate synthetic training data that mirrors the complexity of real-world aviation safety reports. This method significantly improves the model's generalizability and adaptability, enabling rapid updates and customization to meet evolving domain-specific requirements. The development process involved careful data preparation, including the synthesis of entity types and the generation of labeled datasets through template filling. Fine-tuning of the Aviation-BERT model demonstrated strong performance in precision, recall, and F1 scores, showing its efficacy in accurately identifying a wide range of entity types within aviation safety reports. Testing on real-world narratives from the National Transportation Safety Board (NTSB) database highlighted the model's robustness, with performance metrics indicating its potential to significantly enhance the automation of aviation safety report analysis. This work addresses a critical gap in English language NER models for aviation safety and establishes a new benchmark for domain-specific NER systems, promising substantial improvements in the analysis and understanding of aviation safety reports.

Keywords: Natural Language Processing, Text Mining, Aviation Safety, NTSB, ASRS, Named Entity Recognition

1. Introduction

Aviation is considered as one of the safest forms of transport, with 2024 considered one of the safest years on record [1], primarily because of its proactive and continuous focus on safety. The recent trend in aviation safety increasingly gravitates towards data-driven solutions. Aviation safety data has been used in reactive and proactive [2,3] approaches to detect risk factors using different kinds of models. Broadly, a reactive approach to safety focuses on previously occurring accidents and their causal factors to understand historical data. A proactive approach focuses building models that predict the probability of risky events, which can be used to recommend actions.

Natural Language Processing (NLP) is a fast growing area in Artificial Intelligence (AI) where computer models are built to analyze text data. In aviation, the analysis of safety reports, aviation maintenance, and traffic control are the three broad areas where NLP is currently used [4]. For reactive safety analysis, the National Transportation Safety Board (NTSB) Report Database [1] is invaluable for learning lessons from past incidents and accidents. It is an openly available database that stores accident text narratives data in structured and unstructured form. Since any manual analysis of these reports is complicated and time consuming, NLP can serve as a powerful enabler to automatically

[1] https://www.ntsb.gov/Pages/AviationQuery.aspx

 doi:10.20944/preprints202410.0656.v1

extract safety data from accident reports for use in data-driven reactive safety analysis. The models built using these analyses can then be used for improving aviation safety as a whole.

Yang and Huang [5] and Liu [6] conducted a detailed review of applications of NLP techniques in aviation safety and air transportation. The evolution of NLP techniques from word-counting techniques like term-frequency, inverse-document-frequency (TF-IDF), bag-of-words, to long short-term memory (LSTM), to more modern context-capturing transformer-based architectures is visible over the last decade. For instance, Perboli et al. [7] used Word2vec and Doc2vec representations of language for semantic similarity in the identification of human factor causes in aviation. Miyamoto et al. [8] used a bag-of-words approach with TF-IDF to identify operational safety causes behind flight delays using the ASRS database. Zhang et al. [9] used word embeddings and LSTM neural networks for the prognosis of adverse events using the NTSB database.

The state-of-the-art models in NLP use the transformer architecture with the attention mechanism, which allows better capturing of context from textual data [10]. Google AI's Bidirectional Encoder Representations from Transformers (BERT), is one such transformer-based model which revolutionized NLP through its innovative pre-training of deep bidirectional representations from unlabeled text [11]. Recently, transformer-based models are increasingly finding their applications to the aviation-domain. Kierszbaum and Lapasset [12] applied the BERT base model on a small subset of Aviation Safety Reporting System (ASRS) [2] narratives to answer the question "When did the accident happen?". In another work, Kierszbaum et al. [13] pre-trained a Robustly optimized BERT (RoBERTa) model on a limited ASRS dataset to evaluate its performance on natural language understanding (NLU). Recently, Wang et al. [14] developed AviationGPT on open-source LLaMA-2 and Mistral architectures which showed greater performance for aviation-domain specific NLP tasks.

The present work introduces Aviation-BERT-NER, which builds on top of previous aviation domain-specific BERT model developed by Chandra et al. [15]. This tailored variant, Aviation-BERT, incorporated domain-specific data from aviation safety narratives, including those from NTSB and ASRS. By doing so, Aviation-BERT significantly enhanced its performance in aviation text-mining tasks. Aviation-BERT not only captures the unique lexicon and complexities of aviation texts but also surpasses the capabilities of traditional BERT models in extracting valuable insights from aviation safety reports [15]. For instance, Aviation-BERT-Classifiers show improved performance for identifying occurrence categories for NTSB and ASRS text reports [16] compared to the generic English language version of BERT. Aviation-BERT was also shown to outperform other generic English language models more than 4 times its size in expanding Aviation-Knowledge Graph (Aviation-KG) [17]. In a parallel effort by Jing et al. [17], the Aviation-KG question answering system, first proposed by Agarwal et al. [18], is being expanded upon to mine and integrate data from structured knowledge graphs, thereby enabling the system to answer complex questions with accuracy. The development of these knowledge graphs involves a thorough process of mapping out subjects, objects, and their interrelations within texts to form structured data sets. However, this detailed process presents its own set of challenges, particularly in accurately extracting entities and their relationships, which are crucial for the model's effectiveness.

This is where Named Entity Recognition (NER) becomes invaluable, particularly in niche domains like aviation. NER is essential for Knowledge Graph Question Answering (KGQA) systems because it precisely identifies and classifies specific aviation-related named entities, such as airport names, aircraft models, and aviation terms. By doing so, NER can potentially lay a solid foundation for accurately responding to fact-based queries in the aviation field [19]. The rest of this paper is organized as follows: Section 2 reviews the latest in domain-specific NER models, data preparation, related works, and defines the problem; Section 3 details the data preparation while Section 4 provides details

[2] https://asrs.arc.nasa.gov/

 doi:10.20944/preprints202410.0656.v1

for the Aviation-BERT-NER model fine-tuning; Section 5 discusses results while Section 6 concludes the present work.

2. Literature Review and Problem Definition

2.1. Domain Specific NER Models

There are a variety of specialized NER models tailored to specific domains. For instance, developed in 2023, the FiNER-LFs + Snorkel WMV NER model is designed for the financial sector [20] and focuses on entities like names of people, places, and organizations within financial documents, news articles, and regulatory texts, achieving an F1 score of 79.48%. Similarly, the BioClinicalBERT NER model focuses on the healthcare industry, particularly for parsing Electronic Health Records (EHRs) [21]. It is designed to extract essential clinical information related to oncology, such as diseases, symptoms, treatments, tests, medications, procedures, and other specific cancer-related details with an F1 score of 87.6%.

A few NER models have been designed for aviation safety documents. For example, a customized Chinese IDCNN-BiGRU-CRF model created in 2023 targets the extraction of particular entities from aviation safety reports, including event, company, city, operation, date, aircraft type, personnel, flight number, aircraft registration, and aircraft part, with an F1 score of 97.93% [22]. Another Chinese language model, which uses a Multi-feature Fusion Transformer (MFT), achieves an F1 score of 86.10% and focuses on capturing seven entities more related to the space domain [23]. For English-language models, a tailored SpaCy NER model from Singapore identifies and categorizes aviation-specific terms such as airline, aircraft model, aircraft manufacturer, aircraft registration, flight number, departure location, and destination location, achieving an F1 score of 93.73% [24]. Andrade et al. [25] utilize a BERT based model to extract entities from SAFECOM mishap reports pertinent to failure such as modes, causes, and effects, along with control processes and recommendations. On five entities of interest, this model achieves an F1 score of 33% (excluding non-entity terms), with the authors acknowledging the need for additional training data. An aerospace requirements focused model, aeroBERT-NER, is tailored for Model-Based Systems Engineering (MBSE) within the aerospace domain [26] and identifies entities such as system names, organizations, resources, values, and date/time expressions relevant to aerospace with an F1 score of 92%. While this dataset is accessible on Hugging Face, it is important to note the distinct nature of aerospace requirements documents compared to aviation safety reports, as the former does not include specific mentions of aircraft names, airport names, and similar entities. The use of gazetteers and rule-based matching using syntactic-lexical patterns for NER tasks has shown tremendous promise in structured aviation datasets such as Letters of Agreement [27]. However, the performance of this method is unlikely to hold up for aviation safety datasets with a lot of variability, such as ASRS - where each contributor may have a different writing style.

A wide range of generic NER models are publicly accessible via platforms like GitHub and Hugging Face, albeit not all are covered in scholarly articles. For example, the instructNER_ontonotes5_xl model available on Hugging Face is capable of identifying a broad spectrum of entities such as events, facilities, legal documents, geographic locations, products, and services, with an F1-score of 90.77% [28]. Another model found on Hugging Face, the span-marker-bert-base-fewnerd-fine-super, identifies diverse entities that include airports, islands, mountains, transit systems, companies, government agencies etc. with an F1 score of 70.5% [29]. Although these models can assist in preparing datasets by analyzing aviation safety texts to identify specific entities, a process known as pseudo-labeling, the creation of a "Judge Model" still necessitates the use of gold labels or true labels [30]. This implies that manual checking of the data remains critical to correct any errors in labeling, a task that becomes impractical for datasets with more than 1,000 sentences especially when multiple iterations are required.

 doi:10.20944/preprints202410.0656.v1

2.2. Data Preparation Methods

Numerous labeled datasets exist across different domains, including multilingual collections like CoNLL-2003 and OntoNotes, social media compilations such as WNUT-2017, Wikipedia datasets including Wiki Gold, Wiki Coref, and Hyena, as well as biomedical datasets like NCBI Disease, BioCreative, Genia Corpus, BC4CHEMD, JNLPBA, and BC2GM [31]. While there are several labeled datasets for Chinese aviation safety reports, a comparable labeled dataset for English aviation safety reports is found lacking, highlighting the necessity to develop an original aviation safety dataset.

There are several approaches for handling unlabeled datasets in aviation safety reports. One method previously discussed is pseudo-labeling, where models from third parties are employed to analyze aviation safety texts for identifying specific entities. However, developing a Judge Model to oversee this process still demands the utilization of gold or true labels [30]. This necessitates a labor-intensive process of manually checking and correcting misclassified named entities, a task that becomes particularly burdensome with multiple iterations. In addition, adding new types of named entities introduces another layer of complexity, as it requires sourcing models capable of detecting the desired entities, which can be challenging for certain types, like heading or temperature, which may both be stated in degrees and can confuse generic NER models.

Unsupervised NER techniques, which innovate in avoiding manual input, face substantial hurdles when applied to niche datasets such as aviation safety reports [32]. The precision and relevance of seed entities pose a significant challenge in specialized fields like aviation, where the jargon and context diverge markedly from those found in generic or internet-based texts. Moreover, the intricate details and specific characteristics of aviation safety reports might not be adequately tackled by the ambiguity resolution strategies inherent to these systems, resulting in heightened inaccuracies and noise within the extracted entity data.

Another unsupervised approach, CycleNER, utilizes cycle-consistency training alongside seq2seq transformer models, providing an effective strategy for areas with a lack of annotated datasets [33]. Nevertheless, when applied to specific datasets, such as those related to aviation safety, CycleNER might face considerable obstacles. The success of this method depends significantly on the presence of a small, yet highly representative and quality set of entity examples, which may be difficult to procure or excessively niche for aviation safety scenarios. Furthermore, existing challenges within CycleNER, including issues with detecting entity spans and processing sentences devoid of entities, may impede its effectiveness in aviation safety contexts, where precise recognition of technical terminology and intricate entity relationships is paramount.

2.3. Related Work

Template-based strategies, specifically those that utilize structured datasets for NER, seem promising given the challenges presented by various models and data preparation techniques listed earlier. Cui et al. [34] developed a sequence-to-sequence (seq2seq) framework utilizing BART for dynamic template filling, highlighting the importance of context and semantic relationships for accurately predicting and classifying entity spans. However, this approach faces limitations in direct data generation control and requires extensive model familiarity for post-testing adjustments.

The method presented here, inspired by the principles outlined in the dynamic template-filling technique, seeks to exploit the advantages of template-based strategies within NER. By incorporating a deterministic template-filling strategy that leverages structured datasets, the present work overcomes the hurdles noted in the seq2seq framework, such as the challenge of direct control and adaptability in real-world application scenarios. The proposed method not only capitalizes on the structured nature of the data for generating training material but also ensures greater flexibility and ease in making targeted adjustments, enhancing the overall adaptability and effectiveness of NER systems in practical settings.

 doi:10.20944/preprints202410.0656.v1

2.4. Problem Definition

To summarize prior discussion, in the realm of aviation safety, the effective analysis of incident reports is hindered by significant challenges in NER. These challenges include the absence of English-specific NER models in the United States and datasets tailored for aviation safety, the labor-intensive nature of manual data verification for label accuracy, and the complexities inherent in developing a Judge Model for oversight. For unsupervised data generation, the accuracy of seed entities is compromised by the specialized jargon of the aviation field. Dynamic template filling introduces innovative approaches to contextual understanding but lacks direct control over data generation and necessitates substantial expertise for post-testing adjustments. These issues collectively underscore a pressing need for advanced NER solutions that are not only technically sophisticated but also pragmatic and adaptable to the specialized domain of aviation safety.

The present work introduces a novel approach to domain-specific NER model creation applied specifically for the analysis of aviation safety reports. By customizing and extending the concept of template-based NER, this work pioneers several key innovations that address the unique challenges of this domain. These advancements promise not only to enhance the accuracy and efficiency of domain-specific NER systems but also to significantly improve their applicability and responsiveness to real-world needs:

1. **Improved generalizability**: This work enhances template-based NER by directly generating training data from templates populated with diverse entities drawn from structured datasets, reflecting the complexity of aviation safety reports. This ensures the domain-specific synthetic training data accurately represents the multifaceted nature of aviation entities, reducing the risk of overfitting.
2. **Enhanced flexibility and customization**: This work improves flexibility of domain-specific NER systems, enabling rapid updates and customization of entity types and formats to align with specific domain requirements or evolving stakeholder feedback. This adaptability ensures that the NER system remains current and effective, even as aviation safety standards and practices evolve.
3. **Independence from existing models and manual labeling**: The presented methodology eliminates reliance on pre-existing NER models and the intensive labor of manual data labeling. By leveraging structured datasets for template filling, a more efficient, scalable, and error-resistant approach is offered to generating training data, ensuring high-quality inputs from the start.
4. **Ease of adjustments for real-world texts**: In response to testing feedback and real-world application, this system provides swift and precise model adjustments. Whether refining entity definitions or modifying template wording, these changes can be implemented without the need for comprehensive model retraining, facilitating continuous improvement and adaptability to real-world needs.

3. Data Preparation

3.1. Named Entity Data Preparation

Aviation safety reports feature a wide range of named entities and essential key terms vital for thorough analysis. Utilizing narratives from the NTSB database, 17 distinct types of entities were identified, and the corresponding data were compiled from various sources, as shown in Table 1.

 doi:10.20944/preprints202410.0656.v1

Table 1. Synthesized overview of entity types in aviation safety analysis.

Entity Category	Unique Count	Entity Examples	Source
MANUFACTURER	272	'Bell Helicopter', 'Aerospatiale', 'Piaggio', 'Boeing', 'EMBRAER'	flugzeuginfo.net [35]
AIRCRAFT	2,720	'A330-200', '707-100', 'DH-115 Vampire', 'PA38', 'MD11'	
AIRPORT	58,632	'Caffrey Heliport', 'Regina General Hospital Helipad', 'Wilsche Glider Field', '41WA', 'SAN'	GitHub Gist [36]
CITY	15,507	'New Brighton', 'Hortonville', 'Wiesbaden', 'Dimapur', 'Marietta'	
STATE	1,893	'Michigan', 'Sipaliwini District', 'Saarland', 'SO', 'QLD', 'ME'	
COUNTRY	115	'United States of America', 'Ghana', 'South Africa', 'Brazil', 'France'	
AIRLINE	759	'Gulf Air', 'Air Moldova', 'Etihad Airways', 'S7', 'NCA'	Bansard International [37]
DATE	22,115	'in October of 2022 on the 5th', '02-04-2016', '22/07/2016', 'on the 6th of Aug. 2022'	N/A
TIME	14,360	'05:37 CST', '0031 UTC +6', '11:47', '11:56 Central Europe Time', '0945'	
WAY	915	'runway 09', 'rwy 25R', 'taxiway D', 'txwy A4'	
SPEED	800	'120 kts', '107.2 knots', '514.6 mph', '87 miles per hour', '217 km/h'	
ALTITUDE	2,650	'12,422 m', '38,800 feet agl', '10132 ft', 'FL210', '5541 meters'	
DISTANCE	500	'11,981 km', '182.9 miles', '496.7 kilometers', '995 NM', '2000 ft'	
TEMPERATURE	1,500	'9 degree C', '31 Fahrenheit', '-2.5 F', '54 °F', '25 C'	
PRESSURE	900	'101263.6 Pa', '24.2 inches of mercury', '20.5 inHg', '1117 hPa', '1026.2 millibars'	
DURATION	1,100	'35 secs', '11.6 hrs', '1.5 hours', '9 h', '10 mins'	
WEIGHT	700	'56667 kg', '503385 pounds', '1,200 kilograms', '22476.3 lbs', '179.1 tonnes'	
O	N/A	N/A	

The vocabulary for each named entity derived from external sources included a diverse array of forms, such as full names, codes, and abbreviations, reflecting the complexity found in actual aviation safety reports. The counts of unique entities excluded overlapping terms, such as identical abbreviations and codes. This variability also applied to DATE, TIME, and WAY lists, which included the assorted formats seen in genuine reports. Similarly, quantity-based entities (SPEED, ALTITUDE, DISTANCE, TEMPERATURE, PRESSURE, DURATION, and WEIGHT) were represented with realistic ranges and units. Consistency was maintained across entities like AIRPORT, CITY, STATE, and COUNTRY to ensure alignment in the subsequent step of template-based NER dataset generation.

 doi:10.20944/preprints202410.0656.v1

3.2. Template Preparation

To generate templates that closely resemble aviation safety reports, NTSB narratives were manually examined, occasionally adopting their exact structure or making minor alterations in phrasing and sequence to introduce diversity. Named entities within these templates were then replaced with placeholders marked by {}. These placeholders were designed to be randomly filled with named entities from the previously generated lists in Section 3.1. A few examples of such templates are provided below:

- During a routine surveillance flight {DATE} at {TIME}, a {AIRCRAFT_NAME} ({AIRCRAFT_ICAO}) operated by {AIRLINE_NAME} ({AIRLINE_IATA}) experienced technical difficulties shortly after takeoff from {AIRPORT_NAME} ({AIRPORT_SYMBOL}) in {CITY}, {STATE_NAME}, {COUNTRY_NAME}.
- The pilot of a {MANUFACTURER} {AIRCRAFT_ICAO} noted that the wind was about {SPEED} and favored a {DISTANCE} turf runway.
- An unexpected {WEIGHT} shift prompted {AIRCRAFT_NAME}'s crew to reroute via {TAXIWAY} for an emergency inspection.
- {AIRLINE_NAME} encountered unexpected {TEMPERATURE} fluctuations while cruising at {SPEED}, leading to an unscheduled maintenance check upon landing at {AIRPORT_NAME}.
- Visibility challenges due to a low {TEMPERATURE} at {AIRPORT_NAME} led the {AIRCRAFT_NAME} to modify its course to an alternate {TAXIWAY}, under guidance from air traffic control.

Additionally, ChatGPT 4 was utilized to create varied sets of modified templates, with half of them generated by this external model [38]. These templates were generated iteratively using few-shot prompting. The initial prompt included placeholder descriptions for all the entity categories listed in Table 1. The next prompt provided three genuine NTSB narratives to illustrate the structure of aviation safety reports, along with three example templates containing placeholders. Finally, ChatGPT was prompted to create templates with the following instructions:

> *"Using the information provided for entity placeholders, aviation safety reports, and example templates, generate 10 new templates based on aviation safety data, covering a range of possible incidents and scenarios. Do not add placeholders beyond the list already provided. Templates must be single sentences."*

ChatGPT was then prompted to generate further templates in batches of 10, ensuring no repetition of previously generated templates. Each template was reviewed for suitability and added to the collection. To adjust the frequency of specific placeholders, the prompt was modified accordingly. An example of such a prompt is given below:

> *"Generate 10 more templates. Increase the occurrence of placeholders {DURATION}, {WEIGHT}, and {TEMPERATURE}. Do not repeat any previously generated templates."*

This process ultimately resulted in the development of 423 distinct templates (manual and ChatGPT combined), each consisting of a single sentence.

3.3. Labeled Synthetic Dataset Generation for Training

The developed templates were used to construct meaningful sentences by randomly replacing placeholders with named entities. This process was repeated 166 times, resulting in the creation of 70,218 sentences. The precise number of repetitions was determined based on the study outlined in Appendix A, which highlights the flexibility of the template-based approach in generating synthetic training datasets of varying sizes. The entity annotation and BIO tagging were performed simultaneously, with "O" representing words *outside* the entities of interest, "B" marking the *beginning* of an entity phrase, and "I" indicating continuation *inside* the entity phrase. Table 2 provides an example of

 doi:10.20944/preprints202410.0656.v1

placeholders replaced with random named entities, along with their corresponding tags. Additionally, Figure 1 shows the distribution of entities across the training dataset, with a total of 2,125,157 entities (including "O").

 doi:10.20944/preprints202410.0656.v1

Table 2. Example of entity annotation in generated training sentences.

Template	Sentence	Entity	Entity with BIO
During	During	O	O
a	a	O	O
routine	routine	O	O
surveillance	surveillance	O	O
flight	flight	O	O
{DATE}	on	DATE	B-DATE
	the	DATE	I-DATE
	12th	DATE	I-DATE
	of	DATE	I-DATE
	May	DATE	I-DATE
	2012	DATE	I-DATE
at	at	O	O
{TIME}	0821	TIME	B-TIME
,	,	O	O
a	a	O	O
{AIRCRAFT_NAME}	100	AIRCRAFT	B-AIRCRAFT
	King	AIRCRAFT	I-AIRCRAFT
	Air	AIRCRAFT	I-AIRCRAFT
(	(	O	O
{AIRCRAFT_ICAO}	BE10	AIRCRAFT	B-AIRCRAFT
)	)	O	O
operated	operated	O	O
by	by	O	O
{AIRLINE_NAME}	SATA	AIRLINE	B-AIRLINE
	Internacional	AIRLINE	I-AIRLINE
(	(	O	O
{AIRLINE_IATA}	S4	AIRLINE	B-AIRLINE
)	)	O	O
experienced	experienced	O	O
technical	technical	O	O
difficulties	difficulties	O	O
shortly	shortly	O	O
after	after	O	O
takeoff	takeoff	O	O
from	from	O	O
{AIRPORT_NAME}	Andrews	AIRPORT	B-AIRPORT
	University	AIRPORT	I-AIRPORT
	Airpark	AIRPORT	I-AIRPORT
(	(	O	O
{AIRPORT_SYMBOL}	C20	AIRPORT	B-AIRPORT
)	)	O	O
in	in	O	O
{CITY}	Berrien	CITY	B-CITY
	Springs	CITY	I-CITY
,	,	O	O
{STATE_NAME}	Michigan	STATE	B-STATE
,	,	O	O
{COUNTRY_NAME}	United	COUNTRY	B-COUNTRY
	States	COUNTRY	I-COUNTRY
	of	COUNTRY	I-COUNTRY
	America	COUNTRY	I-COUNTRY
.	.	O	O

 doi:10.20944/preprints202410.0656.v1

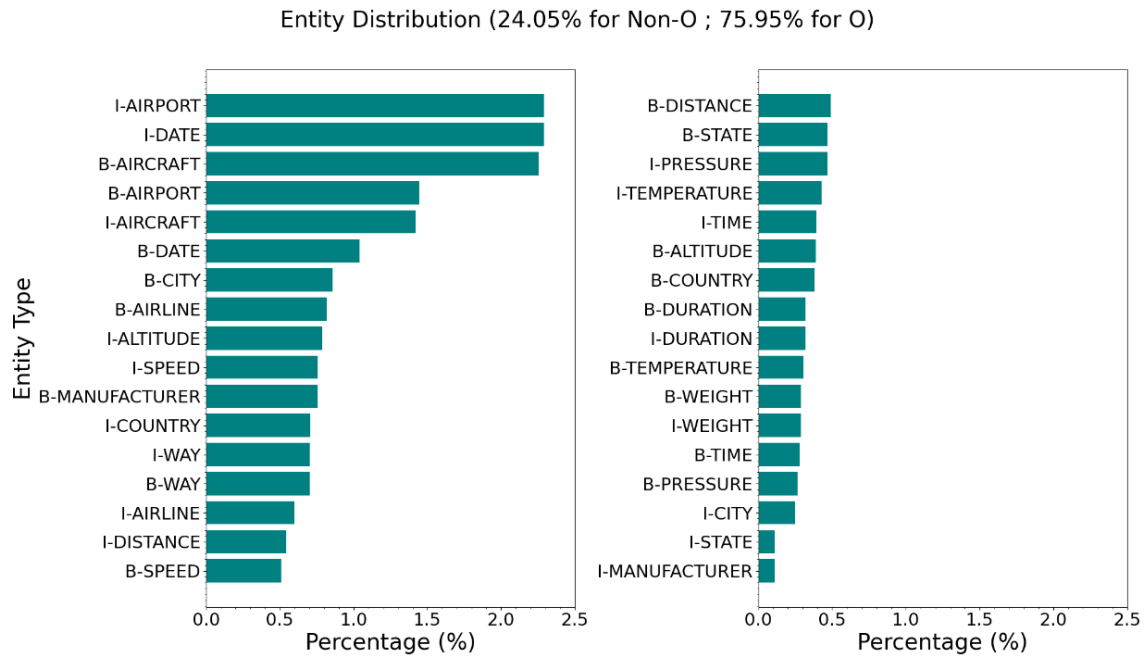


Figure 1. Distribution of entities in synthetic training dataset.

3.4. Labeled Dataset Generation for Testing

To test performance on real-world data, genuine narratives from the NTSB database were considered. As an initial step in the selection process, all narratives in the database were tokenized using the Aviation-BERT tokenizer, and only those narratives with a sequence length shorter than 512 tokens were selected. This was done to prevent indexing errors during inference, as Aviation-BERT is based on the BERT-Base-Uncased model with a maximum allowable sequence length of 512 tokens [11,15]. In total, 108,514 narratives were selected from this process.

From the 108,514 selected narratives, individual narratives were randomly chosen and qualitatively examined for entities of interest. Narratives with a visibly high number and diversity of entities were shortlisted for the test dataset, while others were disregarded. For example, extremely short narratives of one or two sentences with no words within the entities of interest were not considered as candidates for the test dataset. This random selection process continued until 50 narratives were shortlisted. These shortlisted narratives were then manually annotated for all entities with a total of 22,021 (including "O"). Entity distribution for the test dataset is shown in Figure 2.

A secondary test dataset was constructed using the same 50 narratives to understand the effect of shorter inputs on the model's performance. For this purpose, the 50 narratives were broken down into individual sentences without altering the manual annotation for entities. This process resulted in 848 sentences, which can be used to evaluate performance of the same narratives at the sentence level.

 doi:10.20944/preprints202410.0656.v1

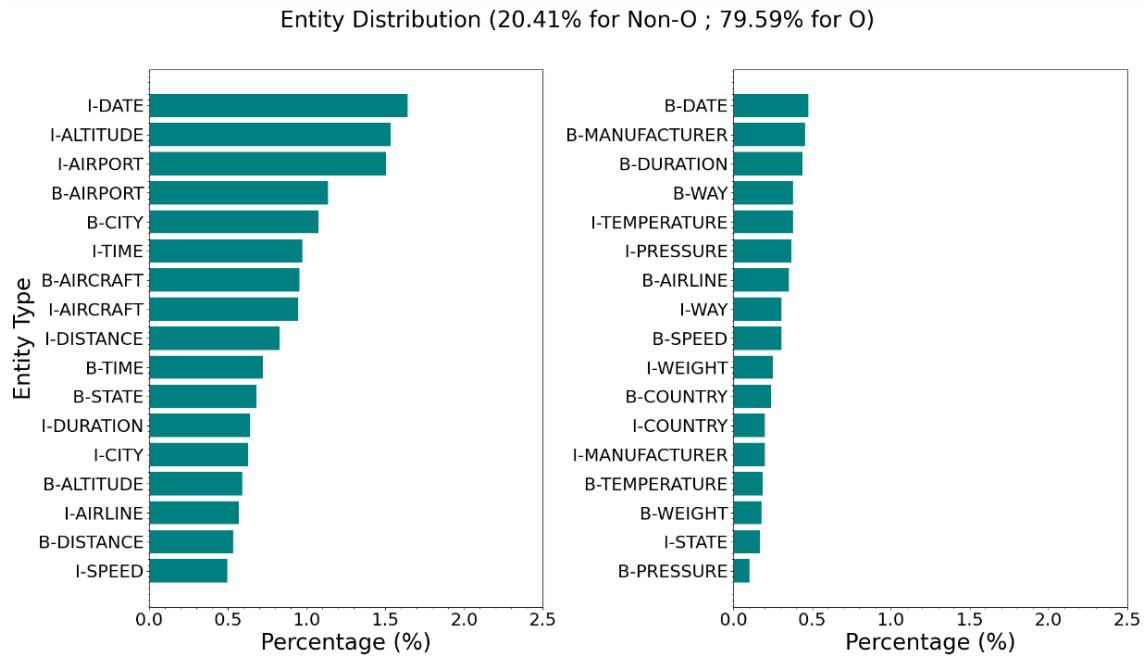


Figure 2. Distribution of entities in test dataset.

4. Fine-Tuning

4.1. Training

The Aviation-BERT model, initially pre-trained on comprehensive textual data from NTSB and ASRS, was fine-tuned for NER using the generated training dataset. The same tokenizer employed during the model's pre-training phase was used for fine-tuning. During fine-tuning, all layers of the Aviation-BERT model were frozen except for the top layer, thereby preserving the model's foundational linguistic knowledge.

The AdamW optimizer [39] was selected for the training stage. A learning rate of 5×10^{-5} and a batch size of 16 were chosen, aligning with best practices for fine-tuning transformer-based models to ensure a balance between quick adaptation to the new domain and maintaining stability in the learning process. Fine-tuning was conducted over three epochs, a duration carefully calibrated to optimize the model's learning without leading to overfitting (see study in Appendix A). A single Tesla V100 32GB GPU was utilized, and fine-tuning was completed in under two hours.

4.2. Testing

After the model was fine-tuned, its performance was evaluated using the 50 test narratives at both the paragraph and sentence levels. The performance was quantified using precision, recall, and F1 scores as metrics. Weighted average values of these metrics were calculated to account for the imbalanced nature of the test datasets. Additionally, the percentages of correct predictions for each entity were evaluated to compare performance at both levels.

The misclassifications and incorrect predictions were then manually reviewed as part of a qualitative assessment. This step was required because the excessive presence of the "O" entity may at times inflate the scores, potentially giving a misleading indication of the model's predictive performance on the actual entities of interest.

5. Results and Discussion

5.1. Scores and Predictions

Precision, recall, and F1 scores for the test narratives at both the paragraph and sentence levels are summarized in Table 3. The percentages of correct predictions for each entity in the test datasets

 doi:10.20944/preprints202410.0656.v1

are presented in Table 4. F1 scores of 94.78% and 95.08% indicate that the model performs well at both levels, with slightly better performance at the sentence level. This is further supported by the higher percentages of correct predictions at the sentence level for nearly all entities. This behavior suggests a clear influence of the type of training data, as the training dataset consisted solely of single sentences derived from templates.

Table 3. Test results.

Test Dataset	Precision	Recall	F1 Score
50 paragraphs	95.34%	94.62%	94.78%
848 sentences	95.59%	94.90%	95.08%

Table 4. Percentages of correct predictions for all entities.

Entity Label	% correct predictions		Entity Label	% correct predictions	
	50 paragraphs	848 sentences		50 paragraphs	848 sentences
B-MANUFACTURER	82.00	**88.00**	I-MANUFACTURER	36.36	**63.64**
B-AIRCRAFT	82.86	**88.10**	I-AIRCRAFT	92.79	92.79
B-AIRPORT	**74.80**	73.60	I-AIRPORT	**92.75**	89.12
B-CITY	**88.14**	86.02	I-CITY	**88.41**	85.51
B-STATE	**96.67**	95.33	I-STATE	91.89	91.89
B-COUNTRY	73.59	73.59	I-COUNTRY	65.91	**70.46**
B-AIRLINE	73.08	**85.90**	I-AIRLINE	75.20	**85.60**
B-DATE	80.00	**87.62**	I-DATE	97.78	**99.72**
B-TIME	93.71	**94.34**	I-TIME	92.06	**97.20**
B-WAY	73.49	**75.90**	I-WAY	88.06	88.06
B-SPEED	88.06	**91.05**	I-SPEED	76.15	**78.90**
B-ALTITUDE	86.15	**90.00**	I-ALTITUDE	69.44	**72.70**
B-DISTANCE	88.89	**93.16**	I-DISTANCE	78.02	**79.12**
B-TEMPERATURE	100.00	100.00	I-TEMPERATURE	90.36	**91.57**
B-PRESSURE	81.82	**95.46**	I-PRESSURE	70.37	**80.25**
B-DURATION	79.38	79.38	I-DURATION	**57.45**	56.74
B-WEIGHT	82.05	**84.62**	I-WEIGHT	60.00	60.00
O	**97.55**	97.35			

The model's performance can also be visualized using a confusion matrix. Figure 3 shows the confusion matrix for the more conservative results at the paragraph level. The darkened diagonal in this figure represents the alignment between predicted and true labels for each entity, with a percentage scale on the right. The prominent diagonal demonstrates the model's strong ability to accurately recognize and categorize the majority of entities.

 doi:10.20944/preprints202410.0656.v1

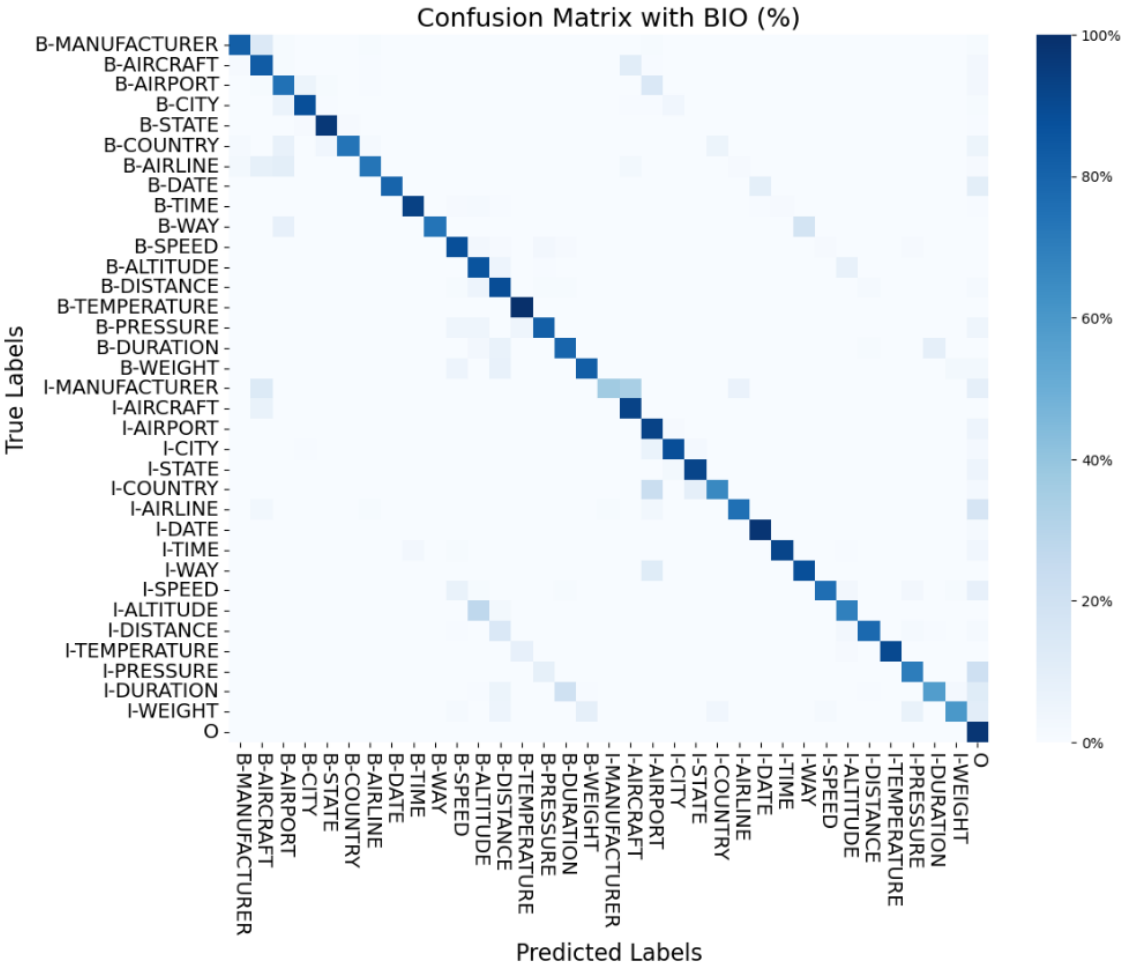


Figure 3. Test confusion matrix for 50 paragraphs.

When adapting the NER model for KGQA systems as in Ref. [17], the focus shifts from the fine-grained boundary delineation typical of BIO (Beginning, Inside, Outside) tagging to a broader emphasis on accurately recognizing and categorizing entities. In the context of KGQA, the detailed entity spans defined by BIO tagging are less relevant, as the primary goal is to understand the intent behind user queries, and directly link identified entities to the knowledge graph. Consequently, this adaptation led to a review of performance metrics without the BIO tagging scheme. The results, summarized in Table 5, show that the model performs even better without the BIO tagging scheme, demonstrating its improved alignment with the operational requirements of KGQA systems.

Table 5. Test results without BIO tagging.

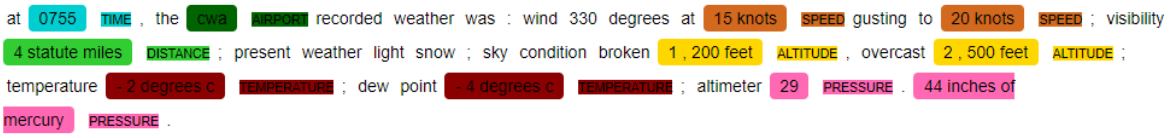
Test Dataset	Precision	Recall	F1 Score
50 paragraphs	96.37%	96.07%	96.14%
848 sentences	96.55%	96.23%	96.32%

As stated in Figure 2, the test dataset contains a relatively high number of "O" entities, comprising nearly 79.59% of all labeled data. While correctly predicting the "O" entity contributes to overall model performance, its excessive influence on the F1 score highlights the need for a qualitative assessment of the results to better understand the model's performance on entities of interest.

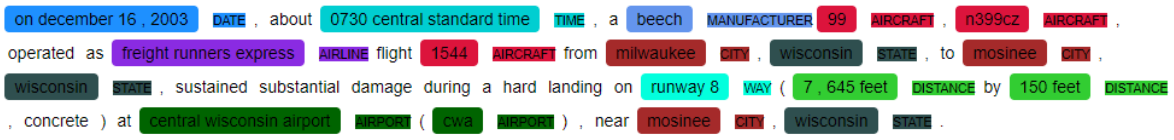
Examples of correct entity classifications and predictions are shown in Figure 4. In Figure 4(a), the model's ability to distinguish between heading angle and temperature is noticeable, even though

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 9 October 2024 doi:10.20944/preprints202410.0656.v1

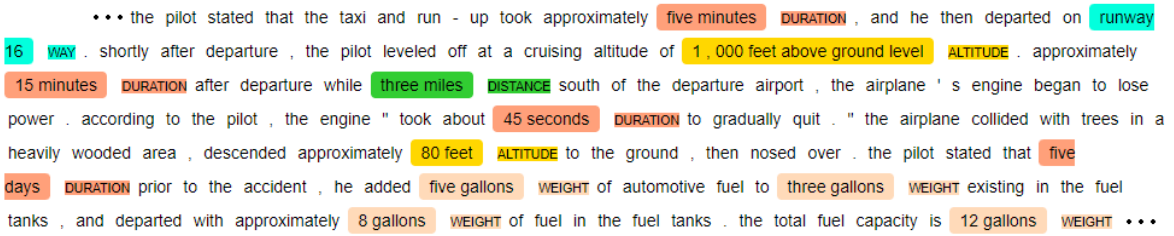
both use the same units. For instance, "330 degrees" is correctly classified as "O" based on meaning and context, while the temperature and dew point are identified as TEMPERATURE. A similar challenge for the model is differentiating between ALTITUDE and DISTANCE. Figure 4(b) illustrates how the model correctly classifies the dimensions of the runway as DISTANCE. Similarly, in Figure 4(c), vertical distance is correctly classified as ALTITUDE. In the same example, quantities in gallons are classified as WEIGHT. While this classification is considered correct for the current model, it can easily be adjusted to volume by introducing a separate entity category during the data preparation and labeling stages.



(a)



(b)

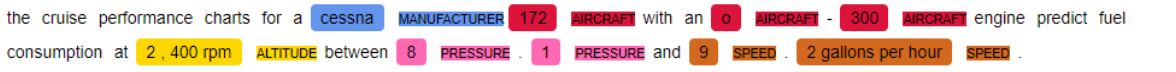


(c)

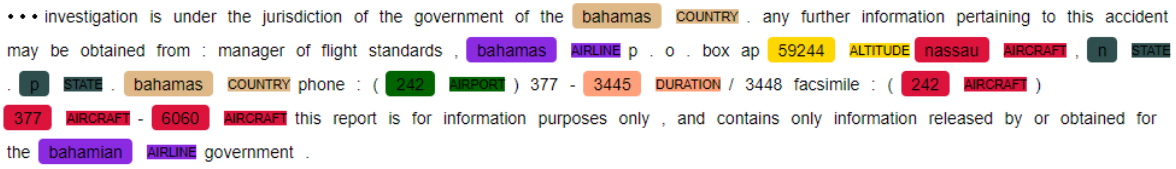
Figure 4. Examples of correct entity classification.

Despite achieving high F1 scores, the model exhibits certain misclassifications, indicating areas for improvement. Figure 5 provides examples of some frequently occurring misclassifications. These misclassifications are particularly evident with aircraft engine specifics, such as model number, fuel consumption, and rotational speed, as shown in Figure 5(a). Another area where the model underperforms is in classifying addresses and telephone numbers as entities outside of interest (Figure 5(b)). These misclassifications likely result from the absence of similar phrases in the training templates labeled as "O", making the model unfamiliar with them. A different yet common type of misclassification is shown in Figure 5(c), where the manufacturer "Beech" is identified as AIRCRAFT. Although this switch from MANUFACTURER to AIRCRAFT negatively impacts the overall scores, it was considered contextually acceptable within the narrative.

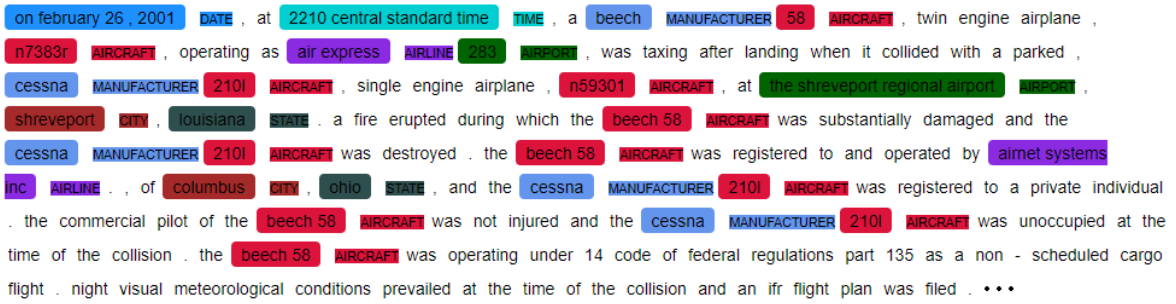
 doi:10.20944/preprints202410.0656.v1



(a)



(b)



(c)

Figure 5. Examples of frequent misclassifications.

The root cause of most misclassifications can be addressed by increasing the diversity of templates to better capture specific non-entity terms. For instance, the strategy of using placeholders can be extended to the "O" label as well. Meaningful templates can be created with placeholders for addresses, telephone numbers, and quantities with units that are not of interest. During dataset generation, random values can be inserted into these placeholders, assigning them the "O" label. This approach will improve the model's ability to recognize entities that fall outside the scope of interest.

 doi:10.20944/preprints202410.0656.v1

Table 6. Comparison of NER models for aviation/aerospace documents.

Model	Focus Area	Language	Identified Entities	F1 Score
Chinese IDCNN-BiGRU-CRF [22]	Aviation Safety Reports	Chinese	1. Aircraft Type, 2. Aircraft Registration, 3. Aircraft Part, 4. Flight Number, 5. Company, 6. Operation, 7. Event, 8. Personnel, 9. City, 10. Date	97.93%
Chinese MFT-BERT [23]	Aerospace Documents	Chinese	1. Companies & Organizations, 2. Airports & Spacecraft Launch Sites, 3. Vehicle Type, 4. Constellations & Satellites, 5. Space Missions & Projects, 6. Scientists & Astronauts, 7. Technology & Equipment	86.10%
Singapore's SpaCy NER [24]	Aviation Safety Reports	English	1. Aircraft Model, 2. Aircraft Registration, 3. Manufacturer, 4. Airline, 5. Flight Number, 6. Departure Location, 7. Destination Location	93.73%
Custom BERT NER model [25]	Aviation Safety Reports	English	1. Failure Mode, 2. Failure Cause, 3. Failure Effect, 4. Control Processes, 5. Recommendations	33% *
aeroBERT-NER [26]	Aerospace Design Requirements	English	1. System, 2. Organization, 3. Resource, 4. Values, 5. Date/Time	92%
Aviation-BERT-NER	Aviation Safety Reports	English	1. Manufacturer, 2. Aircraft, 3. Airport, 4. City, 5. State, 6. Country, 7. Airline, 8. Date, 9. Time, 10. Way, 11. Speed, 12. Altitude, 13. Distance, 14. Temperature, 15. Pressure, 16. Duration, 17. Weight	94.78%

* Score calculation excludes "O" label.

5.2. Model Comparison

In Table 6, Aviation-BERT-NER is compared with alternative models to highlight its capabilities and performance. A direct comparison with Chinese IDCNN-BiGRU-CRF [22], Chinese MFT-BERT [23], and aeroBERT-NER [26] is challenging, primarily because the first two are designed for Chinese language documents, while the latter focuses on aerospace design requirements. Andrade et al. [25] report an F1 score of 33% for their custom BERT NER model, but since the "O" label is excluded from the weighted average calculations, a straightforward comparison with Aviation-BERT-NER may not be possible. They also note that the F1 score achieved is passable for their specific purpose and suggest that additional labeled training data could potentially improve results.

The evaluation of Aviation-BERT-NER, based on a weighted F1 score from 50 narratives of actual aviation safety reports, shows performance closely aligned with Singapore's SpaCy NER [24], though Aviation-BERT-NER identifies 17 entities, compared to just 7 by the prior model. A distinguishing feature of Aviation-BERT-NER is its ability to capture not only aviation-specific entities but also a wide range of quantities. The fine-tuning setup inherently supports the expansion of entity types, allowing for easy refinement of templates and datasets in response to testing outcomes. This flexibility, along with the elimination of extensive manual dataset labeling, makes the model highly practical and adaptable, especially in the evolving landscape of aviation document analysis.

 doi:10.20944/preprints202410.0656.v1

6. Conclusions

Domain-specific NER models serve an important function in the analysis of textual data. This paper presents a novel approach to training such models by generating synthetic training datasets for improved generalizability, flexibility, and customization, while reducing reliance on manual labeling, and facilitating their application to real-world problems.

The Aviation-BERT-NER model enhances the extraction of key terminology from aviation safety reports, such as those found in the NTSB and ASRS databases. A central feature of Aviation-BERT-NER's approach is its use of templates, designed to reflect the diverse formats and scenarios found in real-world aviation safety narratives. This approach not only supports generalizability but also allows for quick adaptation to evolving aviation terminology and standards. The synthetic training dataset can be tailored to accommodate any number of entities while ensuring balance and diversity. In this work, over 70,000 synthetic sentences were generated for training Aviation-BERT-NER to recognize 17 distinct entities. For testing, real NTSB accident narratives were manually annotated to assess the model's performance.

Aviation-BERT-NER addresses common challenges in NER models, such as misclassification, by continuously refining template diversity and strategically enhancing its named entity datasets. This proactive approach to model improvement demonstrates the method's effectiveness in delivering a robust tool for aviation safety analysis. In benchmarking against other NER models, Aviation-BERT-NER outperformed other aviation-specific English NER models while handling more than twice the number of entities. Aviation-BERT-NER achieved an F1 score of 94.78% while identifying 17 entities, compared to Singapore's SpaCy NER, which had an F1 score of 93.73% for 7 entities.

However, the model has limitations in handling entities it was not trained to recognize, such as aircraft engine details, telephone numbers or addresses, which may appear in real-world aviation safety reports. Future work will focus on addressing these gaps. Future work will also involve integrating Aviation-BERT-NER into an Aviation-KGQA system. This integration aims to enhance the KGQA system's ability to formulate accurate query responses by leveraging structured knowledge graphs for detailed information retrieval.

Appendix A

As outlined in Section 3.3, the 423 templates are iteratively processed to generate labeled sentences for the training dataset, with each iteration involving the random selection of entities to populate the templates. The number of iterations must be carefully determined to ensure sufficient model learning without overfitting. This section details the methodology used to select the optimal number of iterations, i.e., the dataset size, while balancing the number of training epochs to achieve the best results.

Table A1 presents models trained with different dataset sizes over 3 to 5 epochs. Note that the dataset sizes are all multiples of 423, as they are generated by iterating the templates. All other aspects, including the hyperparameters identified in Section 4.1, remain consistent for training these models. The paragraph level test dataset containing 50 narratives is used to generate weighted F1 scores for quantitative comparison of performance. Scores without the BIO tagging scheme are also presented for additional comparison.

 doi:10.20944/preprints202410.0656.v1

Table A1. Test results (paragraph level) for different dataset sizes and training epochs.

Model #	Dataset Size	Training Epochs	F1 Score	F1 Score (without BIO)
1	50,337	3	94.45%	95.86%
2	50,337	4	**95.03%**	96.12%
3	54,990	3	93.03%	94.20%
4	54,990	4	94.28%	95.43%
5	60,066	3	93.84%	95.28%
6	60,066	4	94.11%	95.26%
7	60,066	5	93.98%	95.13%
8	65,142	3	94.26%	95.53%
9	65,142	4	93.94%	95.14%
10	70,218	3	94.78%	**96.14%**
11	70,218	4	93.60%	95.19%
12	80,370	3	92.54%	93.65%
13	80,370	4	93.79%	95.20%
14	80,370	5	93.47%	94.69%
15	100,251	3	93.83%	95.47%
16	100,251	4	93.27%	94.82%
17	100,251	5	93.38%	94.68%
18	120,132	3	93.45%	94.60%
19	120,132	4	93.30%	94.49%
20	120,132	5	92.57%	93.87%

Model 2 in Table A1 showed the highest F1 score of 95.03%. While this model might seem like the natural choice, the general trend of high scores across all models being influenced by the "O" entity necessitated a qualitative assessment to evaluate the performance on entities of interest. The high scores already indicated a strong level of correct classifications; therefore, the qualitative assessment focused on the severity of misclassifications. An example of a misclassification observed in Model 2 that disqualified it as the top contender is shown in Figure A1. In some narratives, this model incorrectly identified full stops as CITY, which was deemed an unacceptable error.

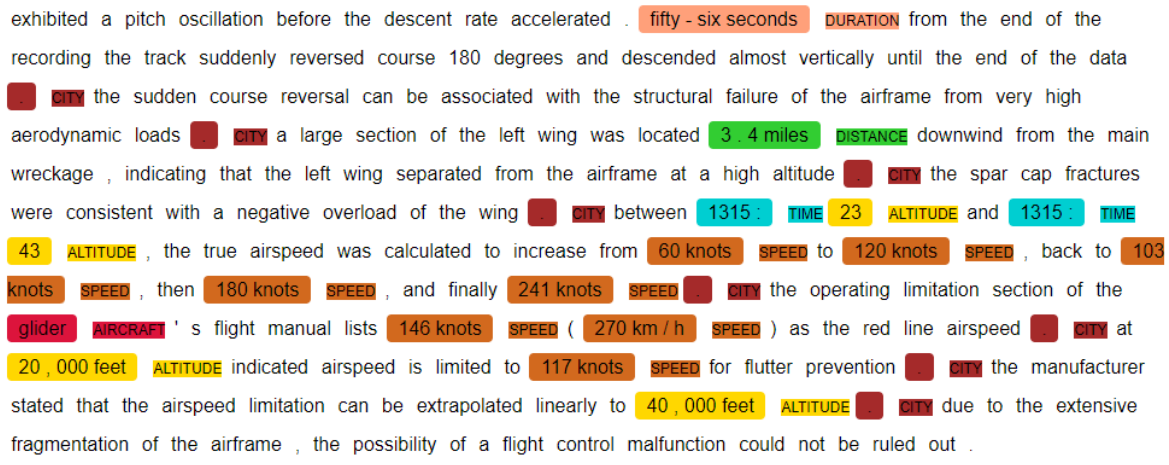


Figure A1. Example of unacceptable misclassification in Model 2.

The next best model, Model 10, with an F1 score of 94.78%, was examined, and an analysis of its common misclassifications is provided in Section 5.1. While Model 10 was not without flaws, it did not exhibit signs of overfitting or serious misclassifications and simultaneously achieved an impressive F1 score. Incidentally, it also achieved the highest F1 score of 96.14% when tested without the BIO tagging scheme. As a result, the dataset size of 70,218, generated through 166 iterations of the templates, and three epochs were selected as optimal for the NER model.

 doi:10.20944/preprints202410.0656.v1

An additional observation can be drawn from the data presented in Table A1. Models trained with larger dataset sizes over a greater number of epochs exhibited reduced performance. This suggests that the model was likely memorizing phraseology and context from the training data, leading to overfitting. Consequently, it can be inferred that merely increasing the volume of templatized data through repeated iterations does not inherently enhance performance.

References

1. International Air Transport Association. IATA Annual Review 2024. retrieved Sept 12, 2024, online: https://www.iata.org/contentassets/c81222d96c9a4e0bb4ff6ced0126f0bb/iata-annual-review-2024.pdf, 2024.
2. Oster Jr, C.V.; Strong, J.S.; Zorn, C.K. Analyzing aviation safety: Problems, challenges, opportunities. *Research in transportation economics* **2013**, *43*, 148–164.
3. Zhang, X.; Mahadevan, S. Bayesian Network Modeling of Accident Investigation Reports for Aviation Safety Assessment. *Reliability Engineering & System Safety* **2021**, *209*, 107371. doi:10.1016/j.ress.2020.107371.
4. Amin, N.; Yother, T.L.; Johnson, M.E.; Rayz, J. Exploration of Natural Language Processing (NLP) applications in aviation. *Collegiate Aviation Review International* **2022**, *40(1), 203-216*.
5. Yang, C.; Huang, C. Natural Language Processing (NLP) in Aviation Safety: Systematic Review of Research and Outlook into the Future. *Aerospace* **2023**, *10*. doi:10.3390/aerospace10070600.
6. Liu, Y. Large language models for air transportation: A critical review. *Journal of the Air Transport Research Society* **2024**, *2*, 100024. doi:https://doi.org/10.1016/j.jatrs.2024.100024.
7. Perboli, G.; Gajetti, M.; Fedorov, S.; Giudice, S.L. Natural Language Processing for the identification of Human factors in aviation accidents causes: An application to the SHEL methodology. *Expert Systems with Applications* **2021**, *186*, 115694. doi:https://doi.org/10.1016/j.eswa.2021.115694.
8. Miyamoto, A.; Bendarkar, M.V.; Mavris, D.N. Natural Language Processing of Aviation Safety Reports to Identify Inefficient Operational Patterns. *Aerospace* **2022**, *9*. doi:10.3390/aerospace9080450.
9. Zhang, X.; Srinivasan, P.; Mahadevan, S. Sequential deep learning from NTSB reports for aviation safety prognosis. *Safety Science* **2021**, *142*, 105390. doi:https://doi.org/10.1016/j.ssci.2021.105390.
10. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *CoRR* **2017**, *abs/1706.03762*, [1706.03762].
11. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 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), 2019, pp. 4171–4186.
12. Kierszbaum, S.; Lapasset, L. Applying Distilled BERT for Question Answering on ASRS Reports. 2020 New Trends in Civil Aviation (NTCA), 2020, pp. 33–38. doi:10.23919/NTCA50409.2020.9291241.
13. Kierszbaum, S.; Klein, T.; Lapasset, L. ASRS-CMFS vs. RoBERTa: Comparing Two Pre-Trained Language Models to Predict Anomalies in Aviation Occurrence Reports with a Low Volume of In-Domain Data Available. *Aerospace* **2022**, *9*. doi:10.3390/aerospace9100591.
14. Wang, L.; Chou, J.; Tien, A.; Zhou, X.; Baumgartner, D. AviationGPT: A Large Language Model for the Aviation Domain. AIAA AVIATION FORUM AND ASCEND 2024. doi:10.2514/6.2024-4250.
15. Chandra, C.; Jing, X.; Bendarkar, M.; Sawant, K.; Elias, L.; Kirby, M.; Mavris, D. Aviation-BERT: A Preliminary Aviation-Specific Natural Language Model. AIAA AVIATION 2023 Forum, 2023. doi:10.2514/6.2023-3436.
16. Jing, X.; Chennakesavan, A.; Chandra, C.; Bendarkar, M.V.; Kirby, M.; Mavris, D.N. BERT for Aviation Text Classification. AIAA AVIATION 2023 Forum, 2023. doi:10.2514/6.2023-3438.
17. Jing, X.; Sawant, K.; Bendarkar, M.V.; Elias, L.R.; Mavris, D. Expanding Aviation Knowledge Graph Using Deep Learning for Safety Analysis. AIAA AVIATION FORUM AND ASCEND 2024, 2024. doi:10.2514/6.2024-4603.
18. Agarwal, A.; Gite, R.; Laddha, S.; Bhattacharyya, P.; Kar, S.; Ekbal, A.; Thind, P.; Zele, R.; Shankar, R. Knowledge Graph - Deep Learning: A Case Study in Question Answering in Aviation Safety Domain. Proceedings of the Thirteenth Language Resources and Evaluation Conference; European Language Resources Association: Marseille, France, 2022; pp. 6260–6270.

19. Mollá, D.; Van Zaanen, M.; Smith, D. Named entity recognition for question answering. Australasian Language Technology Association Workshop. Australasian Language Technology Association, 2006, pp. 51–58.
20. Shah, A.; Gullapalli, A.; Vithani, R.; Galarnyk, M.; Chava, S. FiNER-ORD: Financial Named Entity Recognition Open Research Dataset, 2024, [arXiv:cs.CL/2302.11157].
21. Durango María C., Torres-Silva Ever A., O.D.A. Named Entity Recognition in Electronic Health Records: A Methodological Review. *Healthc Inform Res* **2023**, *29*, 286–300, http://e-hir.org/journal/view.php?number=1175. https://doi.org/10.4258/hir.2023.29.4.286.
22. Wang, X.; Gan, Z.; Xu, Y.; Liu, B.; Zheng, T. Extracting Domain-Specific Chinese Named Entities for Aviation Safety Reports: A Case Study. *Applied Sciences* **2023**, *13*. doi:10.3390/app131911003.
23. Chu, J.; Liu, Y.; Yue, Q.; Zheng, Z.; Han, X. Named entity recognition in aerospace based on multi-feature fusion transformer. *Scientific Reports* **2024**, *14*. doi:10.1038/s41598-023-50705-0.
24. Bharathi, A.; Ramdin, R.; Babu, P.; Menon, V.K.; Jayaramakrishnan, C.; Lakshmikumar, S. A hybrid named entity recognition system for aviation text. *EAI Endorsed Transactions on Scalable Information Systems* **2024**, *11*.
25. Andrade, S.R.; Walsh, H.S. What Went Wrong: A Survey of Wildfire UAS Mishaps through Named Entity Recognition. 2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC), 2022, pp. 1–10. doi:10.1109/DASC55683.2022.9925798.
26. Ray, A.T.; Pinon-Fischer, O.J.; Mavris, D.N.; White, R.T.; Cole, B.F. aeroBERT-NER: Named-Entity Recognition for Aerospace Requirements Engineering using BERT. AIAA SCITECH 2023 Forum, https://arc.aiaa.org/doi/pdf/10.2514/6.2023-2583. https://doi.org/10.2514/6.2023-2583.
27. Pai, R.; Clarke, S.S.; Kalyanam, K.; Zhu, Z. Deep Learning based Modeling and Inference for Extracting Airspace Constraints for Planning. AIAA AVIATION 2022 Forum, 2022. doi:10.2514/6.2022-3755.
28. Aarsen, T. SpanMarker for Named Entity Recognition. retrieved Sept 13, 2024, online: https://github.com/tomaarsen/SpanMarkerNER.
29. Aarsen, T. SpanMarker with bert-base-cased on FewNERD. retrieved Sept 13, 2024, online: https://huggingface.co/tomaarsen/span-marker-bert-base-fewnerd-fine-super.
30. Li, Z.z.; Feng, D.w.; Li, D.s.; Lu, X.c. Learning to select pseudo labels: A semi-supervised method for named entity recognition. *Frontiers of Information Technology & Electronic Engineering* **2020**, *21*, 903–916. doi:10.1631/FITEE.1800743.
31. Jehangir, B.; Radhakrishnan, S.; Agarwal, R. A survey on Named Entity Recognition — datasets, tools, and methodologies. *Natural Language Processing Journal* **2023**, *3*, 100017. https://doi.org/10.1016/j.nlp.2023.100017.
32. Nadeau, D.; Turney, P.D.; Matwin, S. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. Advances in Artificial Intelligence: 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006, Québec City, Québec, Canada, June 7-9, 2006. Proceedings 19. Springer, 2006, pp. 266–277. doi:10.1007/11766247_23.
33. Iovine, A.; Fang, A.; Fetahu, B.; Rokhlenko, O.; Malmasi, S. CycleNER: An unsupervised training approach for named entity recognition. The Web Conference 2022, 2022.
34. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-Based Named Entity Recognition Using BART, 2021, [arXiv:cs.CL/2106.01760].
35. Palt, K. ICAO Aircraft Codes - flugzeuginfo.net. retrieved March 19, 2024, online: https://www.flugzeuginfo.net/table_accodes_en.php, 2019.
36. Gacsal, C. airport-codes.csv - GitHub Gist. retrieved March 19, 2024, online: https://gist.github.com/chrisgacsal/070379c59d25c235baaa88ec61472b28, 2021.
37. Bansard International. Airlines IATA and ICAO Codes Table. retrieved March 19, 2024, online: https://www.bansard.com/sites/default/files/download_documents/Bansard-airlines-codes-IATA-ICAO.xlsx , 2024.

 doi:10.20944/preprints202410.0656.v1

38. OpenAI. ChatGPT (GPT-4). https://openai.com/chatgpt, 2024. Large language model.
39. Loshchilov, I.; Hutter, F. Fixing Weight Decay Regularization in Adam. *CoRR* **2017**, *abs/1711.05101*, [1711.05101].