Whisper-ATC

Open Models for Air Traffic Control Automatic Speech Recognition with Accuracy

van Doorn, Jan; Sun, Junzi; Hoekstra, J.M.; Jonk, Patrick; de Vries, Vincent

Publication date
2024

Document Version
Final published version

Published in
Proceedings International Conference on Research in Air Transportation

Citation (APA)
van Doorn, J., Sun, J., Hoekstra, J. M., Jonk, P., & de Vries, V. (2024). Whisper-ATC: Open Models for Air Traffic Control Automatic Speech Recognition with Accuracy. In E. Neiderman, M. Bourgois, D. Lovell, & H. Fricke (Eds.), *Proceedings International Conference on Research in Air Transportation* Article ICRAT 2024-83

Whisper-ATC: Open Models for Air Traffic Control Automatic Speech Recognition with Accuracy

Jan van Doorn, Junzi Sun*, and Jacco Hoekstra
Faculty of Aerospace Engineering, Delft University of Technology
The Netherlands
*corresponding email: j.sun-1@tudelft.nl

Patrick Jonk and Vincent de Vries
Aerospace Operations Safety and Human Performance
Royal Netherlands Aerospace Centre
Amsterdam, The Netherlands

***Abstract*—Current advancements in machine learning have provided new architectures, such as encoder-decoder transformers, for automatic speech recognition. For generic speech recognition, very high accuracies are already achievable. However, in air traffic control, automatic speech recognition models traditionally rely on domain-specific models constructed from limited training data. This study introduces this newly developed transformer model for air traffic control and provides a set of fully open automatic speech recognition models with high accuracies. This paper demonstrates how a large-scale, weakly supervised automatic speech recognition model, Whisper, is fine-tuned with various air traffic control datasets to improve model performance. We also evaluated the performance of different sizes of Whisper models. In the end, it was possible to achieve word error rates of 13.5% on the ATCO2 dataset and 1.17% on the ATCOSIM dataset with a random split (or 3.88% with speaker split). The study also reveals that fine-tuning with region-specific data can enhance performance by up to 60% in real-world scenarios. Finally, we have open-sourced the code base and the models for future research.**

***Keywords*—Air traffic control, automatic speech recognition, Whisper, machine learning**

I. Introduction

Since the start of Air Traffic Control (ATC) in aviation, voice communication has been a key for pilots' and Air Traffic Controllers' (ATCO) coordination. However, using speech to communicate between the ATCO and the pilot can introduce issues and human errors [1]. The controller often has to manually keep track of the commands given in the transmission, usually by inputting this information into the label of each aircraft on the radar screen, which adds to their workload. Also, outside the operational setting, RT can cause challenges. For example, in safety analysis, where safety analysts transcribe RT in order to study and document incidents. In many use cases, converting the speech to text can reduce the workload of the personnel dealing with the speech audio.

Automatic Speech Recognition (ASR) has been studied in computer science literature, with research dating back to the 1950s [2]. The recent advancements in machine learning have spurred significant progress in developing ASR models. In machine learning, two common approaches exist for developing a learning algorithm. On the one hand, there is *supervised* learning, with long-time-ruling examples such as DeepSpeech [3], [4] and SpeechStew [5]. On the other hand, there is *unsupervised* learning with examples such as Wav2Vec [6], [7] and BigSSL [8]. The difference between those methods is in the labeling of the data. Supervised learning models exclusively rely on labeled data. In the context of training an ASR model, this can lead to limited training data because of the labor-intensive process of creating the labels (i.e., transcribing audio). In contrast, unsupervised learning models operate with unlabeled data, leading to significantly larger volumes of training data [9].

Large supervised learned ASR models are typically trained on around 5,000 hours of labeled training data [5]. While recent unsupervised models can ingest datasets of up to 1,000,000 hours of unlabeled training data [8]. However, neither of the two approaches can be depicted as being the best. A gap existed between small-scale supervised and large-scale unsupervised ASR models. A newly introduced automatic speech recognition model from September 2022 tried to fill this gap. The Whisper model, created by OpenAI, is trained with 680,000 hours of data using weakly supervised learning [10].

Whisper aims to have a robust automatic speech recognition model characterized by high reliability and usability. This is achieved by using a vast amount of diverse training data, which results in broad generalization. This generalization is designed to provide the ability to transcribe out-of-domain audio without a significant drop in performance compared to training data. This is ideal for niche domains with distinct phraseology, such as air traffic control. In addition to the scarce training data and the ability to fine-tune the Whisper model, this makes it an appealing choice for the ATC domain.

ASR in ATC requires a high level of safety assurance, as human performance in speech recognition is very high. In many cases, the current performance of ASR does not match the requirement on robustness, except in use cases like training and simulation of air traffic controllers [11], [12].

This research aims to find out how large-scale, weakly supervised automatic speech recognition could be applied to air traffic control with improved accuracy. In this study, we first explain the Whisper model's potential speech recognition performance in an ATC context. The methodology involved assessing the out-of-the-box performance of Whisper in the ATC domain and determining the possible performance increase that could be reached by fine-tuning Whisper on global and local ATC data.

To stimulate future research and to make these results reproducible, the complete code, all public ASR models, and other needed resources are published in a GitHub repository[1].

[1]Source code is available at https://www.github.com/jlvdoorn/WhisperATC.

II. Data processing

A. Datasets

For this research, two well-known open-source datasets have been used, namely the ATCO2 dataset [13] and the ATCOSIM dataset [14]. This paper focuses on the ATCO2 and ATCOSIM datasets, which contain clean labels and seem suitable for testing machine learning models. These datasets are also frequently used as benchmarks in previous ASR studies [15]. An overview of the datasets' corresponding characteristics can be found in Table I..

TABLE I.: An overview of the used datasets.

	ATCO2	ATCOSIM
Size (hrs)	1.1	10
Region	Europe	Simulation
Speaker	Pilot & Controller	Controller
Language	English	English
Other	Air traffic info	-
Sample rate	16 kHz	32 kHz

The ATCO2 dataset, as part of the ATCO2 project [13], which began in 2020, is a collection of speech from multiple airports, primarily in Europe. The publicly accessible portion that was manually annotated accounts for one hour of speech. It consists of speech collected from pilots and air traffic controllers. Additionally, it is a community-driven project where the audio is captured using simple, very high frequency (VHF) antennas, resulting in relatively noisy data. It also includes air traffic information, such as nearby waypoints and call signs, augmented using the OpenSky network. The audio is captured at a sampling frequency of 16 kHz.

The ATCOSIM dataset comprises ten hours of speech collected during simulated sessions at the Eurocontrol experimental center. It exclusively features speech from the controller role, specifically from ten ATCOs in the en-route position. Since it is a simulated session, the data clarity is significantly higher. Furthermore, the audio is captured at a higher sampling frequency of 32 kHz, which, coupled with the low noise footprint, results in superior audio quality [14] compared to ATCO2.

B. Pre-Processing

Our first step involves processing the transcripts. As previously mentioned, the transcripts must be standardized into a single format since both datasets are not created equally. The actual text is extracted from extensible markup language files and purified by removing text within brackets, e.g., [...], (...), and $<$ ... $>$. Furthermore, for the ATCO2 dataset, radar data are converted into a simple machine-readable format. Lastly, we filter out all audio and transcript pairs where the transcript is empty (i.e., corresponding to empty audio files).

Once the audio and corresponding labels are processed, the dataset is divided into two parts. A portion of the data serves as training data, specifically for the fine-tuning of Whisper. The remaining portion is used for validation during Whisper's fine-tuning process. The dataset is typically divided into an 80% training and 20% validation data split. These proportions are used randomly to generate the dataset's training and respective validation portions. Upon creation, the datasets are uploaded to the HuggingFace Hub, a development platform and repository system for machine learning models. For performance assessment beyond the baseline, only the validation split is utilized for evaluation.

The final datasets, as detailed in Table II., are structured as follows: The ATCO2 dataset includes 446 training samples and 113 validation samples. Each sample comprises an audio file, a transcript, and an additional file containing radar data corresponding to the audio. The ATCOSIM dataset consists of 7,646 training samples and 1,913 validation samples, each including an audio file and a transcript but without any radar data files.

TABLE II.: An overview of the pre-processed datasets.

	ATCO2	ATCOSIM
Total Size (hours)	1.1	10.46
Train Size (hours)	0.86	8.37
Validation Size (hours)	0.23	2.09
Total Samples	559	9,559
Train Samples	446	7,646
Validation Samples	113	1,913

C. Evaluation Metrics

The most common evaluation metric in automatic speech recognition is the WER (Word Error Rate) [16]. It compares the transcript with the reference (ground truth) supplied by the dataset on a per-word basis. The following formula can calculate the WER:

$$WER(\%) = \frac{S + I + D}{N} \times 100\%. \quad (1)$$

Here, S represents the number of replaced words, I is the number of incorrectly inserted words, and D is the number of words missing in the transcript compared to the reference. Conversely, N represents the total number of words in the reference. The WER (Word Error Rate) score is commonly expressed as a percentage.

The WER score is calculated by simultaneously comparing all generated transcriptions against all references. We apply a weighted average method by evaluating the WER across the entire set of transcriptions rather than per utterance. This approach ensures that utterances with fewer words carry a different weight in the overall calculation than those with more words. Consequently, each transcribed word is given equal significance.

III. Transformer-based Whisper models

A. Whisper model

The Whisper model is one of the few early models that was made open-source by OpenAI. The series of models (including different model sizes) combines both supervised and unsupervised learning methodologies by leveraging a large dataset composed of 680,000 hours of data. This dataset contains a variety of audio sources, including multiple languages, dialects, and acoustic environments.

The model is a transformer-based neural network model, which can capture extended contextual information better than other auto-regressive models. The weakly supervised training leverages data that is less accurately labeled or exact than in fully supervised learning, such as subtitled audio from YouTube.

B. Zero-shot capability

Whisper is trained with a large amount of data from quite diverse sources. It can already be applied to ATC speech without additional training (also known as zero-shot learning). This paper examines such zero-shot learning capability by applying different sizes of Whisper models directly to ATC speech data. The word error rate is then compared to the models that are further trained (fine-tuned) with ATC-specific voice data.

C. Prompting

Prompting is a free text input feature of the Whisper model that can improve the quality of transcripts by making it more likely to correctly transcribe context-specific words and produce transcripts in the desired style. This can be done by providing acronyms that are likely to be named in the audio, by providing transcripts of previous segments if an audio file is split into multiple segments, or by providing a segment of text with the desired style in punctuation [17].

During the construction of the prompting scheme, the first step is to state the context (i.e., "air traffic control communications"). This is then extended with a list of airlines (e.g., KLM, Lufthansa, Speedbird). Subsequently, location-specific items such as full call signs, waypoints, and entities are added (e.g., KLM Six Eight One, WOODY, Amsterdam Radar). Ultimately, the NATO alphabet and domain-specific vocabulary, such as ILS and VFR (Visual Flight Rules), are incorporated into the prompt.

D. Normalization

Whisper was developed in such a way that it predicts the raw transcripts in the training set. Because of that, it tends to produce naturalistic transcriptions in terms of, for example, punctuation. However, because the datasets in this study are all in (different) specific formats, it is necessary to perform an additional processing step in order to be able to compare the generated transcripts with the labels. This process is what we call *normalization*.

Normalization is applied in the post-processing phase before comparing the word error rate against the manual labels. For example, it converts *speed bird* (output from Whisper) into *BA*.

Whisper includes a built-in normalizer that serves as a foundation for the normalization process and is adapted to specific ATC-specific terminology and phraseology. It is enhanced with several functions and filters:

1) It ensures that only text is passed through (i.e., removing any non-alphanumeric characters). It then ensures that all numbers are represented numerically and splits numbers into individual digits (e.g., 501 becomes 5 0 1).
2) It standardizes certain wordings (e.g., "goodbye" becomes "good bye") to maintain a consistent format.
3) It processes everything as lowercase text. [2]

In Figure 1, the prompting and normalization steps are illustrated during the inference. Audio and prompt are first provided as inputs for the Whisper model. Then, the output is further processed with the normalizer to provide a consistent text format for word error rate evaluation.

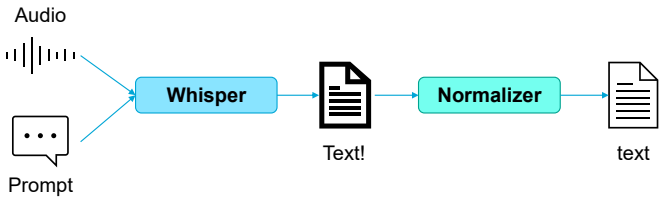

Figure 1. Application of prompting and normalization in voice transcribing

IV. Fine-Tuning of Whisper Models

The HuggingFace *transformers* Python package is used for fine-tuning [18]. This Python library makes the model and dataset easily shareable on the HuggingFace platform, providing a streamlined pipeline for fine-tuning any open model hosted on the HuggingFace Hub. To test their performance, the Whisper models are fine-tuned on the *ATCO2* and *ATCOSIM* datasets. The fine-tuning process of the Whisper model can be summarized as follows:

1) In the data processing step, all audio files are resampled at 16 kHz, aligning with the required format.
2) The Whisper tokenizer transforms the labels into input tokens. These tokens represent the words in the lexicon of the Whisper language model.
3) Whisper's feature extractor is used to calculate the log-mel spectrogram, expressed as an array of input features (e.g., an array containing the signal's power for each frequency at a given time point).
4) A data collator is constructed to ensure the input tokens and the spectrograms are the same length in time. In cases of silence in the audio, it is expressed as a special input token to ensure no information mismatch between the inputs and the labels.
5) The loss function, based on the word error score, is used for training.

During the fine-tuning process, samples in the training split are loaded in a predefined batch size to be utilized in parallel, improving training efficiency. During training, the parameters (weights) of the Whisper models are adjusted based on the difference between the predicted output and the labels from this specific dataset. The fine-tuning process cycles over the training dataset a fixed number of times (epochs). After fine-tuning, the models are uploaded to the HuggingFace Hub[3].

[2]The detailed normalizer is implemented at https://github.com/jlvdoorn/WhisperATC/blob/main/Evaluate/Normalizer.py.

[3]Trained models are openly shared at https://huggingface.co/jlvdoorn.

A. Hyperparameters

Each dataset underwent its fine-tuning process, requiring dedicated hyperparameters due to differences in dataset size. Table III. lists the parameters used for fine-tuning each dataset. We decided to train each model for approximately 100 epochs due to time constraints on the shared GPU (Graphical Processing Unit) resources. The rest of the hyperparameters were also chosen accordingly.

TABLE III.: The hyperparameters used in each fine-tuning process.

Parameter	ATCO2	ATCOSIM
Max. Steps	2,800	12,500
Train Samples	446	7,646
Batch Size	16	64
Epochs	∼100	∼100
Eval Steps	100	1,000
Save Steps	100	2,000

B. Hardware Resources

Fine-tuning the Whisper models requires high-performance GPU hardware. The Delft High-Performance Computing cluster (DHPC) from the Delft University of Technology [19] provides shared high-performance GPUs for the model training in this study.

Our WhisperATC models are fine-tuned using one NVIDIA A100 graphics card with 80 GB of video memory. Additionally, one AMD EPYC 7402 24C CPU and 128 GB of working memory are used. This configuration resulted in a fine-tuning time of approximately one day per model.

C. Inference

During the inference process, the performances of both the original Whisper models and the fine-tuned WhisperATC models are assessed. Each audio file in the validation dataset is processed in the same manner as during training.

The inference is conducted twice for each audio file: once with the prompt and once without the prompt. The prompt initially includes the words *Air Traffic Control Communications* to indicate the general context. It is then extended with specific terminology (e.g., ILS (Instrument Landing System), knots, heading) and the NATO alphabet. In the case of ATCO2, radar information is also added to the prompt.

As the language of ATC speech is English, we manually set the language to English during inference. We also tested cases where the language is not manually set, comparing the performance of the inferences.

The word error rate is calculated after each file has been transcribed twice (with and without prompting). The word error rate is evaluated twice: the first time with normalizing both the reference and the transcript, and the second time with previously defined normalization applied.

V. Results

A. Performance of zero-shot, normalization, and prompting

Firstly, we aim to compare the prediction performance of the base Whisper models without any fine-tuning to establish a baseline for comparison. In this analysis, the performance of the base Whisper Large v2 model on the ATCO2 and ATCOSIM datasets is presented in Table IV.. Furthermore, the results are also visualized in Figure 2.

TABLE IV.: The base Whisper Large v2 performance on ATCO2 and ATCOSIM datasets, each with and without prompting and normalization during the inference.

Norm	Prompt	ATCO2	ATCOSIM
no	no	71.62%	79.11%
yes	no	29.05%	17.98%
yes	yes	24.03%	16.74%

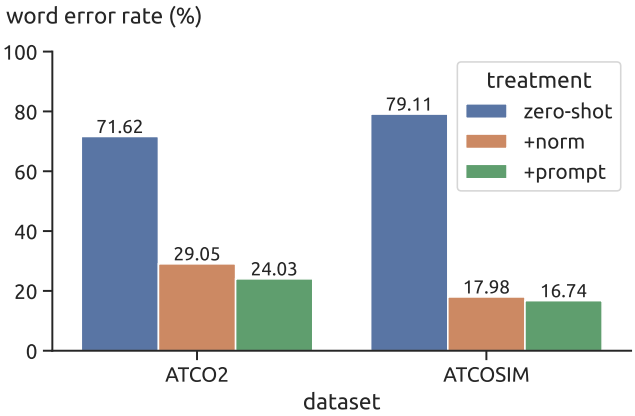

Figure 2. The performance improvement introduced by normalization and prompting.

We can see that the base model, without normalization and prompting, yields a word error rate of around 72% and 79% for the ATCO2 and ATCOSIM datasets, respectively.

With normalization, which converts natural language to ATC-specific transcriptions, we observe a significant reduction in error rate to around 29% for ATCO2 and 18% for ATCOSIM. With prompting, the error rates further decrease to 24% for ATCO2 and 17% for ATCOSIM.

B. Fine-Tuned Model Performance

To analyze the improvement of fine-tuning with different datasets, we created three fine-tuned models as follows:

1) WhisperATC A2: Fine-tuned with ATCO2 data only.
2) WhisperATC AS: Fine-tuned with ATCOSIM data only.
3) WhisperATC A2-AS: Fine-tuned with both ATCO2 and ATCOSIM data.

With both prompting and normalization, we evaluated the performance of these three models on the ATCO2 and ATCOSIM datasets separately. Note that the evaluations apply to validation data only.

The following Table V. shows the word error rates of different fine-tuned models on both datasets. We observe that model performance generally improves with the fine-tuning process.

The only exception occurs when we fine-tune the model with ATCOSIM data but apply it to the ATCO2 dataset; in this case, the word error rate increases. This is likely because the model becomes overly specific for higher quality audios with fewer

TABLE V.: The performance of the base and fine-tuned models on the ATCO2 and ATCOSIM datasets

Model	ATCO2 data	ATCOSIM data
base model	24.03%	16.74%
A2	14.66%	15.84%
AS	34.34%	1.19%
A2-AS	13.46%	1.17%

speakers from ATCOSIM. The performance of these different models is also illustrated in Figure 3.

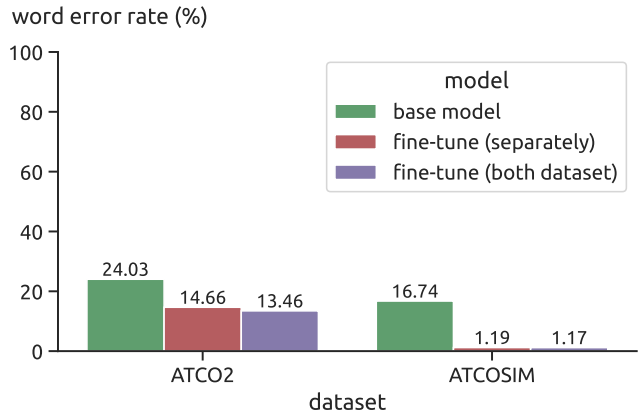

Figure 3. The performance of the blank and fine-tuned models on the ATCO2 and ATCOSIM datasets.

Examining the results, fine-tuning Whisper on the ATCO2 dataset resulted in a WER score of 14.66% on the same dataset. Additional training of the model on the ATCOSIM dataset led to a slight performance improvement, with a WER of 13.46

Conversely, fine-tuning Whisper on the ATCOSIM dataset resulted in a more significant WER reduction. The blank Whisper model achieved a 16.74% WER on the ATCOSIM dataset. By fine-tuning Whisper with the 8.37 hours of training data in the ATCOSIM dataset, the WER dropped to just 1.19%. Extending the training with an additional 0.86 hours from the ATCO2 dataset reduced the WER to 1.17%.

The impact of fine-tuning becomes evident in specific examples. For instance, the blank model predicted: "Telegraph, Skyfinal 25 for touch and go," whereas the fine-tuned Whisper model accurately predicted: "Hotel Echo X-ray final two five for touch and go." The fine-tuned model produces meaningful sentences instead of incorrectly predicting some words. While the blank model recognized some *terminology*, fine-tuning enabled the model to be trained with specific *phraseology*.

C. Performance of Different WhisperATC Models

To evaluate the performance of ATC speech transcription across different sizes of Whisper models, we initially fine-tuned individual models with ATCOSIM and ATCO2 datasets separately. We then compared the word error rates of the validation datasets from these models. The evaluation also included variations with and without manually setting the language to English. In total, 18 rounds of fine-tuning were conducted.

Furthermore, we applied the *normalization* post-processing step to the prediction outputs to generate a new set of normalized outputs aligned with air traffic control phraseology. We then further evaluated the improved word error rates.

Table VI. presents the results with the corresponding datasets and fine-tuned models. The column *WER (raw)* shows the word error rates directly from the fine-tuned models, while the column *WER (norm)* displays the word error rates with the normalization process applied.

TABLE VI.: Overview of transcription performances from different WhisperATC model sizes

Dataset	Model	WER
ATCOSIM	tiny	30.25%
	tiny.en	2.27%
	base	3.49%
	base.en	2.56%
	small	1.50%
	small.en	1.75%
	medium	17.13%
	medium.en	1.20%
	large-v2	1.19%
ATCO2	tiny	126.88%
	tiny.en	74.46%
	base	36.16%
	base.en	47.4%
	small	22.79%
	small.en	42.16%
	medium	17.99%
	medium.en	22.75%
	large-v2	14.66%

For these results, we observe that prediction performance is generally higher for the ATCOSIM dataset than the ATCO2 dataset. This difference is primarily attributed to audio quality, as ATCOSIM voices are collected in a controlled simulation environment, whereas ATCO2 voices are collected over VHF radio frequency with low-cost receivers. Consequently, the audio quality is significantly lower in the ATCO2 dataset.

Within each dataset, varying performances are also evident. Specifically, for the ATCOSIM dataset, the word error rates are lower when models specific to the English language are used. However, for the ATCO2 dataset, manually setting the language to English diminishes transcription performance. This could be due to the ATCO2 dataset featuring a diverse range of speakers, including both controllers and pilots, from various geographic regions.

An anomaly is noted in Table VI.: the performance for the fine-tuned medium-sized model on the ATCOSIM dataset shows an unreasonably large error. Since its fine-tuning dataset, procedure, and hardware are consistent with other models, we hypothesize that there might be an issue with the base Whisper medium model.

Overall, we find that with simulated ATC speech data, like ATCOSIM, the optimal performance (with fine-tuning and normalization) is around a 1.2% word error rate, for real-life aggregated voice data from different regions, like ATCO2, the lowest word error rate is approximately 18%. These results are comparable to, or even surpass, the performance of state-of-the-art but closed-source models.

VI. Verification with Local Air Traffic Speech

Data from LVNL (Air Traffic Control, the Netherlands) are utilized to verify the effectiveness of our methodology in real-world operations. The blank Whisper model is first employed to establish a baseline on the LVNL dataset. Subsequently, we fine-tune the Whisper model with the LVNL dataset to assess performance improvement.

A. LVNL Dataset

The provided data encompasses a week of audio recordings (3rd Oct. 2022 - 9th Oct. 2022) from tower controllers. The audio files vary in length and include speech from both controllers and pilots. Notably, the dataset lacks metadata such as speaker IDs. As the recordings originate from the controller's side, the audio exhibits relatively high signal noise for pilot speech. Additionally, it is captured at a sampling frequency of only 8 kHz, lower than other datasets discussed earlier. Only the audio is supplied, necessitating the creation of transcripts.

The first step involves selecting a manageable subset of the LVNL data (139 hours across 50,000 files) for evaluation. Ultimately, approximately three hours of audio are manually labeled.

The audio is resampled from 8 kHz to 16 kHz, and empty audio files, i.e., files without utterances, are removed. The files are then shuffled randomly to eliminate time-based (and thus speaker-based) biases.

Labeling the LVNL data, a labor-intensive task is facilitated by using Prodigy[20]. This tool aids in labeling by providing pseudo-labels from a pre-trained voice-to-text model. The provided text is manually edited while listening to the audio, resulting in accurate LVNL dataset labels. Some audio files are rejected during labeling due to excessive noise, non-English speech, or lack of utterances.

The final dataset comprises three hours of audio with approximately 1000 files. This dataset is randomly divided into training and validation partitions using an 80% to 20% ratio, resulting in 799 training samples (2 hours and 23 minutes) and 202 validation samples (37 minutes).

Performance assessment on the LVNL audio follows the same procedure as the baseline performance assessment on the ATCO2 and ATCOSIM datasets. The WER is calculated in four configurations of prompting and normalization. Due to data ownership, the LVNL data processing is confined to specialized NLR (Royal Netherlands Aerospace Centre) hardware. The fine-tuning process is executed on a server with an NVIDIA Tesla V100 GPU with 32GB of VRAM.

1) LVNL Baseline Performance: First, the blank Whisper model is evaluated on the created LVNL dataset. The word error rate is compared with its performance on the ATCO2 and ATCOSIM datasets. The results are presented in Table VII..

For the untrained base Whisper model, the best score of 32.02% word error rate is achieved on the LVNL dataset with normalization.

2) LVNL Fine-Tuned Performance: Figure 5 illustrates the fine-tuning process of the Whisper model on the LVNL dataset alongside the ATCO2 and ATCOSIM datasets.

TABLE VII.: The performance of the blank Whisper model on the ATCO2, ATCOSIM, and the LVNL dataset. The first two columns indicate whether normalization and prompting are used.

Norm	Prompt	ATCO2	ATCOSIM	LVNL
no	no	71.62	79.11	78.49
yes	no	29.05	17.98	32.02
yes	yes	24.03	16.74	35.09

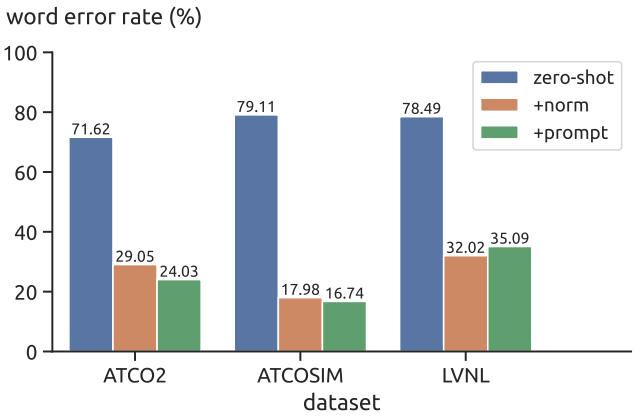

Figure 4. The performance improvement introduced by normalization and prompting.

The blank Whisper model established a baseline WER of 32.02% on the LVNL dataset. Fine-tuning on the 2 hours and 23 minutes of the training set resulted in a WER of 13.28%, translating into a relative reduction of 59% due to fine-tuning, outperforming the improvement with the model trained on ATCO2 data.

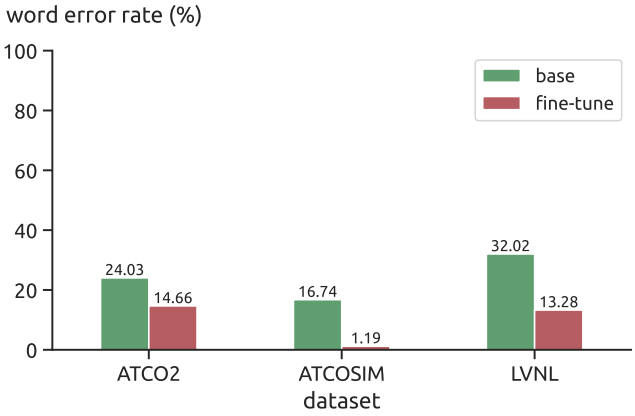

Figure 5. The effect of fine-tuning and evaluating on the difference datasets. Prompting and normalization are used.

The fine-tuning effect can be observed in the following example:

- Blank model prediction: "Total green warfare elevator level four VHF final eight right."
- Fine-tuned model prediction: "Tower, good morning, KLM eight zero four; we are final eighteen, right."

This is likely due to accents and non-English speech used. With the fine-tuned model, even though it is not perfect (e.g., "we

is" should be "we are at"), it yields far better results than the blank model. The latter transcript contains meaningful content, whereas the blank model produced nonsensical text.

In comparison, the reduction in word error rate is even more pronounced with the ATCOSIM dataset. The blank Whisper model reached a WER of 16.74%, transcribing one in six words incorrectly, which is already relatively low. With fine-tuning on the ATCOSIM dataset, it achieved a word error rate of merely 1%, transcribing only one in 85 words incorrectly and a relative reduction of 93%.

VII. Discussion

A. Performance Comparisons

The performance of ATCO2 data and LVNL data are similar. Moreover, the error rate is higher than ACTOSIM. This difference is attributed to real-world data containing a higher noise level and a lower sampling frequency. In contrast, the ATCOSIM dataset was produced in a simulated environment with direct audio capture, resulting in clearer audio.

The fine-tuned models achieved a WER of 13.46% on ATCO2 and 1.17% on ATCOSIM, surpassing the previous state-of-the-art benchmarks of 15.4% on ATCO2 and 5.0% on ATCOSIM [13], [15]. This indicates that WhisperATC sets a new standard for ASR on these datasets.

When considering the required training data, Whisper's efficiency becomes even more apparent. The previous ATCO2 benchmark utilized 3600 hours of training data from the complete ATCO2 dataset [13]. In contrast, training Whisper on just 0.86 hours from the ATCO2 training partition yielded a WER of 14.66% on the ATCO2 validation set. Extending training to include ATCO2 and ATCOSIM datasets (totaling 9.23 hours) achieved the WER of 13.46%. This suggests that the fine-tuned Whisper model outperforms the best available model on the ATCO2 dataset with significantly less training data.

A similar trend is observed for the ATCOSIM dataset, where the best-performing model required 176.4 hours of training data, substantially more than the hours needed for the ATCOSIM model in our case. Again, 80% of the ATCOSIM dataset was used for training, and validation was performed on the remaining 20%.

B. Training and Testing Split for ATCOSIM Dataset

For the ATCOSIM dataset, which consists of recordings from only ten speakers and includes speaker labels, splitting the dataset based on speaker IDs could be beneficial for a more rigorous evaluation. In order to do so, speakers *sm2* and *zf2* were assigned to the validation set and the rest to the training set. The fine-tuned model produced an error rate of 3.88% on the new validation set. Although higher than the previously mentioned 1.17%, it still establishes a new benchmark compared to the current state-of-the-art of 5.0%.

C. Out-of-Domain Voice Data for Training

The ATCO2 project concluded that using standard speech corpora (e.g., LibriSpeech[21], CommonVoice[22]) for training an ASR model does not effectively enhance speech recognition performance in the ATC context [13], [23]. However, likely due to a difference in scale in both the (pre-)training dataset and model size, transfer learning was shown to be an effective method for improving speech recognition performance in ATC.

D. Effect of Prompting and Normalization

In this study, we implemented prompting and normalization processes. Both techniques significantly reduce the word error rate. However, it is important to note that prompting relies on a priori knowledge, which may only sometimes be feasible in real-time transcription. Nevertheless, it could be implemented using a data augmentation system, such as incorporating radar data. In contrast, based on a posteriori knowledge, normalization is more feasibly applied after transcribing.

E. Future Work and Applications

We observed that the ATCO2 and ATCOSIM fine-tuned ASR models achieved up to 13.46% WER on ATCO2 and 1.17% WER on ATCOSIM. Further fine-tuning on local data (ANSP dataset) resulted in a WER of 13.28%.

Multiple studies indicate the feasibility of ASR application in the ATC domain [24], [25]. One of the potential practical applications of ASR in air traffic control is training and simulation [11], [12], where the required performance is less critical than in real-world operations. The accuracy of WhisperATC, mainly based on ATCOSIM, suggests its potential in simulations.

The current model's performance is sufficient for applications with lower performance requirements, like post-processing audio for incident analysis. ASR could be used to transcribe audio for text-based investigations and documenting purposes.

A more challenging next step would be extracting events such as callsigns, take-off clearances, etc. This could be used in different applications such as operational analysis tools or even as an additional safety net. In order to do so, real-time processing may be necessary.

F. Data Limitation

Accessing suitable data for training an ASR model in the ATC domain is challenging. The ATCO2 and ATCOSIM datasets are the only free publicly available datasets of adequate quality, yet they are relatively small (1 hour and 10 hours, respectively). The lack of data is a commonly cited issue in this field [16], [13]. More accessible, public data availability is crucial for improving ATC-related ASR models and promoting open research.

Data diversification is also crucial. Ideally, training data would encompass recordings from across the globe, incorporating various accents, phraseologies, and ATCo positions. However, the ATCO2 and ATCOSIM datasets are limited mainly to European data, with minimal diversity. This results in models fine-tuning on these datasets and performing less effectively on data not included in the training set. This is a common issue in machine learning that can be addressed by training on more diverse data. Ultimately, this could eliminate the need for local dataset fine-tuning, as demonstrated in this research.

VIII. Conclusion

This research focuses on applying automatic speech recognition to air traffic control using Whisper, a large-scale, weakly supervised ASR model. The main objective was to evaluate the performance of fine-tuned Whisper models for transcribing air traffic control speech.

From the baseline evaluation, it is clear that blank Whisper performs relatively well on the ATCO2 and ATCOSIM datasets, with WERs of 24% and 17%, respectively, indicating a good understanding of ATC speech. The fine-tuning significantly enhances performance, setting new state-of-the-art WERs of 14% on ATCO2 and 1.2% (or 3.9 when split by the speaker) on ATCOSIM. The model's ability to comprehend ATC-specific vocabulary and phraseology is particularly noteworthy. Further, validation with actual ANSP voice data demonstrates the potential application of Whisper in real-world ATC environments.

A significant contribution of this paper is the provision of the first open models for air traffic control speech recognition with such accuracy. Transformer-based models like WhisperATC have the performance to begin reducing air traffic controllers' workload in the short term. In the long term, they hold significant potential to transform air traffic control operations.

Acknowledgment

The authors would like to acknowledge LVNL for providing data and their views on the topic. In addition, the authors acknowledge the use of computational resources of the DelftBlue supercomputer, provided by Delft High-Performance Computing Centre (https://www.tudelft.nl/dhpc).

References

[1] R. Wever, G. v. Es, and M. Verbeek, "Air-ground communication safety study causes and recommendations," tech. rep., Eurocontrol, 2006.

[2] R. Pieraccini, "From AUDREY to Siri: Is speech recognition a solved problem?," 2012.

[3] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep Speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 12 2014.

[4] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos, E. Elsen, J. Engel, L. Fan, C. Fougner, T. Han, A. Hannun, B. Jun, P. LeGresley, L. Lin, S. Narang, A. Ng, S. Ozair, R. Prenger, J. Raiman, S. Satheesh, D. Seetapun, S. Sengupta, Y. Wang, Z. Wang, C. Wang, B. Xiao, D. Yogatama, J. Zhan, and Z. Zhu, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," *33rd International Conference on Machine Learning, ICML 2016*, vol. 1, pp. 312–321, 12 2015.

[5] W. Chan, D. Park, C. Lee, Y. Zhang, Q. Le, and M. Norouzi, "SpeechStew: Simply Mix All Available Speech Recognition Data to Train One Large Neural Network," *arXiv preprint arXiv:2104.02133*, 4 2021.

[6] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised Pre-training for Speech Recognition," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2019-September, pp. 3465–3469, International Speech Communication Association, 4 2019.

[7] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *Advances in Neural Information Processing Systems*, vol. 2020-December, 6 2020.

[8] Y. Zhang, D. S. Park, W. Han, J. Qin, A. Gulati, J. Shor, A. Jansen, Y. Xu, Y. Huang, S. Wang, Z. Zhou, B. Li, M. Ma, W. Chan, J. Yu, Y. Wang, L. Cao, K. C. Sim, B. Ramabhadran, T. N. Sainath, F. Beaufays, Z. Chen, Q. V. Le, C.-C. Chiu, R. Pang, and Y. Wu, "BigSSL: Exploring the Frontier of Large-Scale Semi-Supervised Learning for Automatic Speech Recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1519–1532, 2022.

[9] J. Delua, "Supervised vs. Unsupervised Learning: What's the Difference?," 2021.

[10] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," 2022.

[11] J.-P. Imbert, H. Christophe, and J. Yannick, "Think different: how we completely changed the visualization of Pseudo-Pilots," in *Graphics Interface*, (Halifax), pp. 257–264, Canadian Information Processing Society, 6 2015.

[12] R. Tarakan, K. Baldwin, and N. Rozen, "An Automated Simulation Pilot Capability to Support Advanced Air Traffic Controller Training," in *The 26th Congress of ICAS and 8th AIAA ATIO*, (Anchorage, Alaska), American Institute of Aeronautics and Astronautics, 9 2008.

[13] J. Zuluaga-Gomez, K. Veselý, I. Szöke, P. Motlicek, M. Kocour, M. Rigault, K. Choukri, A. Prasad, S. S. Sarfjoo, I. Nigmatulina, C. Cevenini, P. Kolčárek, A. Tart, and J. Černocký, "ATCO2 corpus: A Large-Scale Dataset for Research on Automatic Speech Recognition and Natural Language Understanding of Air Traffic Control Communications," *arXiv*, 11 2022.

[14] K. Hofbauer, S. Petrik, and H. Hering, "The ATCOSIM Corpus of Non-Prompted Clean Air Traffic Control Speech.," in *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, (Marrakech, Morocco), European Language Resources Association (ELRA), 1 2008.

[15] J. Zuluaga-Gomez, P. Motlicek, Q. Zhan, K. Vesely, and R. Braun, "Automatic Speech Recognition Benchmark for Air-Traffic Communications," in *Proc. Interspeech 2020*, (Shanghai), pp. 2297–2301, 10 2020.

[16] S. Badrinath and H. Balakrishnan, "Automatic Speech Recognition for Air Traffic Control Communications," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2676, pp. 798–810, 1 2022.

[17] OpenAI, "OpenAI docs speech-to-text." https://platform.openai.com/docs/guides/speech-to-text/prompting, 2008. [Online; accessed 09-February-2024].

[18] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-Art Natural Language Processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, (Online), pp. 38–45, Association for Computational Linguistics, 10 2020.

[19] Delft High Performance Computing Centre (DHPC), "DelftBlue Supercomputer (Phase 1)." https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1, 2022.

[20] I. Montani and M. Honnibal, "Prodigy: A modern and scriptable annotation tool for creating training data for machine learning models."

[21] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 4 2015.

[22] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common Voice: A Massively-Multilingual Speech Corpus," in *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC 2020)*, pp. 4211–4215, 2020.

[23] J. Zuluaga-Gomez, K. Veselý, A. Blatt, P. Motlicek, D. Klakow, A. Tart, I. Szöke, A. Prasad, S. Sarfjoo, P. Kolčárek, M. Kocour, H. H. Honzačernocký, C. Cevenini, K. Choukri, M. Rigault, and F. Landis, "Automatic Call Sign Detection: Matching Air Surveillance Data with Air Traffic Spoken Communications," *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 59, p. 14, 12 2020.

[24] J. Zuluaga-Gomez, A. Prasad, I. Nigmatulina, P. Motlicek, and M. Kleinert, "A Virtual Simulation-Pilot Agent for Training of Air Traffic Controllers," *Aerospace*, vol. 10, p. 490, 5 2023.

[25] Y. Lin, L. Deng, Z. Chen, X. Wu, J. Zhang, and B. Yang, "A Real-Time ATC Safety Monitoring Framework Using a Deep Learning Approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, pp. 4572–4581, 11 2020.