Automatic Speech Recognition and its Contextual Enhancement for Singapore ATC Voice Communication

Jayakrishnan Melur Madhathil
Institute for Infocomm Research(I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
jayakrishnan_melur_madhathil@i2r.a-star.edu.sg

Nguyen Ngoc Khanh
Institute for Infocomm Research (I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
nguyen_ngoc_khanh@i2r.a-star.edu.sg

Lee Seounghoon
Institute for Infocomm Research(I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
lee_seounghoon@i2r.a-star.edu.sg

Tran Anh Dung
Institute for Infocomm Research(I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
tanhdung@i2r.a-star.edu.sg

Luong Trung Tuan
Institute for Infocomm Research (I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
luong-tt@i2r.a-star.edu.sg

Tran Huy Dat
Institute for Infocomm Research (I^{2}R)
Agency for Science, Technology and Research (A*STAR)
Singapore
hdtran@i2r.a-star.edu.sg

***Abstract*: In a first for Singapore Air Traffic Control (ATC), a complete pipeline of Automatic Speech Recognition (ASR) of voice communication between pilots and Air Traffic Controllers (ATCOs) is presented. Increased complexity due to multi-accented speech, cockpit noise, and speaker dependent biases were overcome by using data sufficiently large enough for training the models, collected across multiple domains namely enroute, approach and tower. We also carried out detailed benchmarking and analysis of various ASR technologies ranging from hybrid HMM-DNN to supervised End to End (E2E) to pre-trained semi-supervised models fine-tuned with ATC voice data. This benchmarking helped us to conclude that traditional hybrid HMM-DNN is still competitive enough to be used in domain-specific areas like ATC. We enhanced the Callsign Recognition Rate (CRR) from audio, with a fast, efficient method, significantly improving it. The preprocessing pipeline includes our cutting-edge Voice Activity Detection (VAD), Speaker Turn Detection, and Speaker Role Detection (SRD) pipeline. We achieved a WER of 5.48%, in addition to improving the CRR by 6.01%.**

Keywords: Air Traffic Control, Callsign Recognition, Air Traffic Controller, Automatic Speech Recognition, Contextual Speech Recognition.

I. Introduction

ASR for ATC is an integral part of the Air Traffic Management (ATM) system. It plays an important role in reducing the workload of ATCOs (or controllers in short) with the ever-increasing air traffic. Even though the ASR technology has matured in generic natural language speech transcription, it remains a challenge in ATC ASR due to the noisy voice channel. ASR systems in the ATC domain demand high accuracy since they are directly related to the safety of the aircraft and the people.

We have collaborated with the Civil Aviation Authority of Singapore (CAAS) to obtain ATC communication speech data from Singapore's Changi Airport. The total amount of audio data after removing silence comes to about 400 hours, which was labelled by a third-party vendor.

In the first part of this work, we focus on training baseline ASR models and fine-tuning pre-trained models with the same data. Two models were trained from scratch: the first being a Kaldi [1] based HMM-DNN model and the second, a joint Connectionist Temporal Classification (CTC)-Attention based End-to-End (E2E) model using the Wenet [2] toolkit. We also fine-tuned a Wav2vec2 conformer ASR pre-trained model with 400 hours of labelled CAAS data. We have benchmarked on the CAAS test data provided and found that the Kaldi-based HMM-DNN model is still up to the mark.

In the second part of the work, the focus is on improving the CRR. We employ two methods for this purpose. The first method is to exclusively build a contextual Language Model (LM) from the augmented training text corpus. The corpus is prepared in such a way that we first identify the callsigns from the text and replace them with the contextual callsigns. The

contextual LM is then trained from this augmented corpus and is used in the ASR decoding as usual. Since this method involves frequent re-training of the LM, it is not suitable for real-time use. In the second method, the attempt is to use the n-best from the lattice and search for a possible match for the callsign among it. Since both the trained hybrid and E2E models are lattice- generating models, it is suitable as a baseline for the second method. In this, the model is supplied with contextual callsign information along with audio data. The callsign is an important information in the voice communication between the pilot and ATCO. An ATCO talks to different pilots during a certain period using the same frequency, and they use callsigns to differentiate between the pilots. It is hence very important for recognizing these callsigns correctly. The contextual callsign information is obtained from either surveillance data or flight plan data. Our key contributions to this paper are as follows:

- First comprehensive ASR system solution for Singapore ATC
- Efficient preprocessing pipeline for ASR systems
- Benchmarking ASR Engines against popular pre-trained models
- Simple and effective offline and online solutions for improved callsign detection by Contextual Speech Recognition (CSR)

II. RELATED WORKS

A. ATC ASR

Work in this direction started as early as 2012, as in [3], where a system for transcribing ATC voice data in natural language is presented. The system in [4] focuses on assistance-based speech recognition. Recent works like [5] use more complex architectures like TDNNF and CNN+TDNNF. There have also been efforts to use pre-trained ASRs like Wav2Vec2 [6] and Whisper [7] models, by fine-tuning them with limited supervised ATC speech data. However, all these works are based on the publicly available datasets such as ATCO2[8], ATCOSIM [9], UWB-ATCC [10], which only covers European Airspace. Unfortunately, the models trained on these datasets fail to perform well with the accented Singaporean English speech. The main reason is the lack of labelled public data for Singapore ATC voice communications. LiveATC [11] provides access to data pertaining to Singapore airspace, but there are no labels, and the recorded audio is mostly very noisy. We have provided decoding results of the test set mentioned in section III-C, results and discussions, in Table-1 by using an open-source ATC ASR model from [12], to substantiate our point.

B. Contextual Speech Recognition(CSR)

There were many attempts earlier to use contextual callsigns for improving the callsign recognition rate in speech data. In [13], the authors have used HCLG and lattice boosting via FST composition. They have assumed that contextual information is specific to each utterance, which is very difficult to obtain in practice. In [14], authors trained a BERT-based model to predict the callsigns from the transcribed text. As an alternative to lattice-based contextual boosting, [15] suggest a method suitable for online implementation in graphics processing units (GPUs). A real-time system that predicts a sequence of likely commands in the transcribed speech is implemented in [16].

Section III gives an overview of the ATC ASR system including the Voice Activity Detection (VAD), Speaker Turn Detection, Speaker Role Detection, and baseline ASR systems used and detail the experimental set up and discusses the ASR results. Section IV gives details of the ways to include the contextual information into the ASR system, namely the Contextual Language Model (CLM) method and n-best matching method, with experimental set up and results. Section V concludes the work, with possible future work for improving the system.

III. ATC ASR SYSTEM OVERVIEW

Figure 1 shows the full pipeline of the ATC ASR system.

A. VAD, Speaker Turn Detection(STD),Speaker Role Detection(SRD)

An energy-based VAD is used to determine the silence or speech segments in the captured voice data. ATCO - pilot communication is usually done through a push-to-talk (PTT) mechanism. The impulse signal generated while pressing the PTT button is captured and filtered. This short-duration signal helps to determine the boundary of pilot-ATCO speech. The method is very simple and effective for CAAS data. A joint STD and SRD approach is presented in [16], by combining a BERT model with VAD by chunking ASR transcripts.

Segmented speech segments after VAD and speaker turn detection should be identified as pilot or ATCO segments. Traditionally, the diarization approach [17] is used to detect such a turning point. As we know, the good diarization approach [18] still has a relatively high diarization error rate. Meanwhile, such continuous (without a silence interval) pilot and ATCO segments do not occur frequently during ATC communication. From the spectrogram of the CAAS data in Figure 3, we observed that there is a noticeable difference in the high- frequency range. Highlighted ellipsoids demonstrate that the high-frequency parts in the pilot are heavily attenuated compared to that of the controller. We exploit this difference to identify the segments by using the following condition:

$$\begin{aligned} &\mathit{Frequency\ band\ ratio\ of\ each\ speech\ segment} \\ &\qquad = \frac{\mathit{low\ frquency\ band\ energy}}{\mathit{high\ frequency\ band\ energy}} \end{aligned}$$

Here, the low-frequency band is from 200Hz to 2800Hz, while the high-frequency band is from 2800Hz to 3400Hz. If the

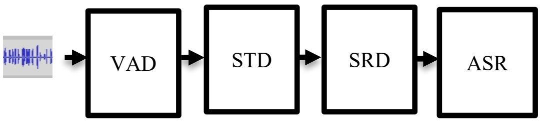

Figure 1. ASR system overview showing preprocessing steps, i.e., Voice Activity Detection (VAD), Speaker Turn Detection (STD) and Speaker Role Detection (SRD)

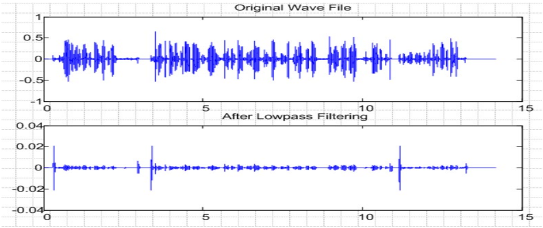

Figure 2. A typical wav file before and after applying low-pass filter with cut-off of 100Hz.

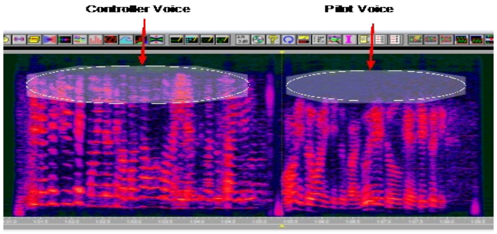

Figure 3. A typical controller and pilot speech spectrogram.

ratio is relatively small, then the segment is classified as a controller segment; otherwise, it is a pilot segment.

Overall, this method of segmentation and speaker role detection achieves a 94% clustering rate on the CAAS dataset, based on the annotated segment labels.

B. *Baseline ASR*

The audio was recorded at the receiver site, which has access to both pilot and controller radio transmissions for most frequencies in the tower, approach, and enroute domains. The audio was then transcribed by a third-party vendor. Raw recordings contain long stretches of silence. So, to get 400 hours of speech data, at least 4000 hours of recordings were required. The audio is then processed for separating into pilot and controller segments, using a speaker turn detection module. It then passed through a speaker role detection module to determine to which speaker each segment belongs to.

- *Kaldi Based Hybrid HMM-DNN Model*

  In this benchmark, we used a standard hybrid approach from [19] (LF-MMI/Chain model) as a baseline for the evaluated ASR models. Generally, it consists of 2 parts: an Acoustic Model (AM) and a Language Model (LM). The AM is a TDNN-F [20] which is used to predict a posterior distribution over the tied Hidden Markov Model (HMM) states corresponding to context-dependent phonemes. These posterior distributions are then combined with a pronunciation dictionary (i.e., the lexicon) and a n-gram LM to construct a search graph in the form of WFST [21]. During the inference, the decoding is done via the beam search, which looks for the best paths in the constructed graph.

- *Wenet Based E2E joint CTC-Attention Model*

  E2E ASR models have replaced the traditional Kaldi based hybrid HMM-DNN models in almost all domains. One advantage of the E2E ASR model is its ability to model inter-word level dependency, making Language Models optional in the ASR system. Moreover, CTC models don't need frame-level alignments, unlike HMM-DNN models. They can also be trained on massive amounts of data, due to its larger number of parameters. Here, we have chosen a joint CTC-Attention model from the Wenet [2] toolkit. Audio Encoder follows conformer architecture, while decoding is attention based.

C. *Experimental Set up*

- *Hybrid HMM-DNN set up*
  Lexicon: The lexicon was updated to include all words found in the new transcribed audio data corpus, which consisted of 400 hours of silence-reduced audio from all three ATC domains. Additionally, the lexicon was updated to include all the new callsigns provided by CAAS.
  Language Model: We used a 3-gram language model trained with text transcripts of training set.
  Acoustic Model: The system utilizes 40-dimensional MFCC features and 100-dimensional features from i-vectors as input. A 13-layer TDNN-F [19] was employed with 1280 dimensions for each hidden layer and 128 dimensions for the linear bottleneck. Data augmentation techniques [22] including speech reverberation (3 times) and speech perturbation (0.9 to 1.3) were applied to increase the amount of training data and improve the robustness of the ASR system. The model was trained with audio samples having a sampling rate of 8KHz.

- *E2E ASR Set up*
  The baseline E2E ASR is a joint CTC/Attention model, with the encoder following a conformer [23] architecture consisting of 8 attention heads, 12 encoder blocks, and 2048 linear units. The decoder architecture is of bi-transformer type with 8 attention heads, 3 decoder blocks, and 2048 linear units. Spectral Augmentation [24] and Speed Perturbation are utilized as data augmentation techniques. The tokenizer used is Byte Pair Encoding (BPE) with a dictionary size of 5000, trained with the training text. We have not employed a language model. The beam search algorithm used is CTC prefix beam search

with a default beam size equal to 10. A fast attention rescoring is used to select the final 1-best. The input feature is a filter bank with number of bins equal to 80, with a frame size of 25 msec and a 10 msec frame shift. In the hybrid CTC/attention set up, the default CTC weight is set to 0.3. The E2E model has 121 million parameters and was trained with audio samples with a sampling rate of 8KHz. Spectral Augmentation is chosen as the data augmentation method. The model is trained with 400 hours of CAAS data and 500 hours of Librispeech.

- *Fine-tuning Wav2vec2 Conformer Model*
  The base model chosen for fine-tuning is Wav2Vec2-Conformer-Large-960h-ft from Huggingface [25]. Wav2Vec2 Conformer is based on the Wav2Vec2 [26] architecture with attention blocks replaced by Conformer [23] blocks. The fine-tuning data consists of CAAS 400 hours data. A batch size of 2 was used, with a total of approximately 162k samples. The total steps were 6000k and the learning rate was set to 1e-5. Nvidia A40 GPU (4 cards, 46 GB each) was used for fine-tuning. Since the base model is trained on 16KHz samples, the original 8KHz samples were up sampled to 16KHz before fine-tuning.

- *Results and Discussions*
  Test Set: The test set was provided by CAAS, recorded during the period from June 13th to 18th June 2023, for a duration of 1 hour each from 2 PM to 3 PM. The audio data provided is from channels 123.7 MHz and 133.8 MHz. Silence parts from the long audio are removed using the VAD described earlier. Each speech segment is decoded using the respective ASR models as shown in Table-I.

  Table-I shows that, when fine-tuned with region-specific data and with LM, pre-trained models perform better than Kaldi and E2E models. However, Wav2Vec2 model has more than 10 times the parameters of the Kaldi HMM-DNN model. The E2E model, with 121 million parameters, can accept large amounts of data. However, we found that the performance of the E2E model with LM is not stable. If we want to add new words into the acoustics, then the model must be re-trained with additional audio data, which could be difficult to obtain. Kaldi models are flexible in this regard, as we can force it to understand new words by adding those into the lexicon and training with additional text data with lots of occurrences of the new words.

TABLE-I FILE BASED WER RESULTS

Filename	Open-source model [12]	HMM-DNN	E2E	Wav2vec2 (with LM)
123.7MHz-13-06-2023	31.39	5.00	5.18	4.24
123.7MHz-14-06-2023	28.22	8.05	6.8	7.22
123.7MHz-15-06-2023	30.27	5.87	6.43	6.02
123.7MHz-16-06-2023	28.5	8.26	7.14	7.59
123.7MHz-17-06-2023	31.02	4.87	5.68	4.87
123.7MHz-18-06-2023	31.2	6.5	5.64	7.14
133.8MHz-13-06-2023	29.36	4.98	6.97	5.32
133.8MHz-14-06-2023	26.41	4.62	3.17	3.03
133.8MHz-15-06-2023	30.74	5.20	4.65	4.51
133.8MHz-16-06-2023	28.69	4.58	5.60	5.09
133.8MHz-17-06-2023	27.57	2.88	4.64	3.03
133.8MHz-18-06-2023	31.67	6.54	6.72	7.59
Average	**29.59**	**5.61**	**5.72**	**5.48**

IV. ATC CONTEXTUAL SPEECH RECOGNITION

A callsign is a unique identifier for each aircraft used by the ATCOs to address a specific aircraft. It follows a standard format set by the International Civil Aviation Organization (ICAO). The first part is airline name followed by a unique number.

Callsign	**Spoken Callsign**
SIA807	SINGAPORE EIGHT ZERO SEVEN
CPA797	CATHAY SEVEN NINER SEVEN

Pilots & ATCOs use spoken callsign during voice communication, which follows ICAO phraseology. The flight information including callsign can be obtained even before the arrival (through flight plan data) or immediately after the flight's entry into the airspace (through surveillance information from radar). Typically, this contextual information can be used to improve the recognition performance of the ASR.

The source of contextual information and the voice recording system are two separate systems, as shown in Figure 4, which are synchronized by means of time stamps (TS). The source could be an Automatic Dependent Surveillance-Broadcast (ADS-B) radar system or a system capable of predicting contextual information from flight plan data. The contextual information (including the callsigns) can be utilized by the ASR system to improve its performance, provided the audio and the contextual information are synchronized by means of timestamps.

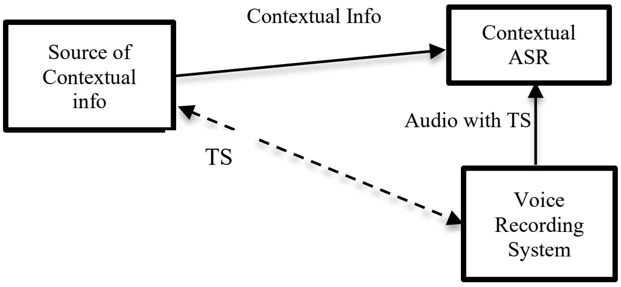

Figure 4. Contextual information source and voice recording system.

Now, how often the source could pass the contextual information depends on the implementation, but generally, it is difficult to synchronize both audio and contextual information at each utterance level. In our approach, we removed this constraint and developed a simple method that depends on the contextual information provided over a fixed window of 1 hour. This method can be applied to all lattice-generating ASRs. We also implemented a second method for callsign recognition as an offline solution. It is similar to the method described in [14]. Here we are restricting the search space of the Weighted Finite State Transducer (WFST), by using a Contextual Language Model (CLM), which is a tri-gram language model trained with the training text corpus after replacing the original callsigns with the contextual callsigns. Figure-5 shows the details of this approach.

A. *Using Contextual Language Model (CLM) for decoding*

The contextual callsigns can be incorporated into the ASR system in two ways: at the pre- and post-recognition phases. In the pre-recognition phase, we alter the weights of the model, so that the search space is limited, while in the post-recognition phase, we match the contextual callsigns within the n-best sequences, hopefully finding a matching callsign within each sequence. The N-best approach works only if the callsign of interest is in the lattice. We propose both methods here, the former one as an offline solution and the n-best matching as a real-time solution.

The newly constructed contextual LM is employed to decode new utterances containing callsigns that may not have been encountered by the acoustic model during its training. First, a contextual callsign list, obtained from flight plan data, is prepared for the duration under consideration. The callsigns in the training text are then identified and are replaced by contextual callsigns. This new text corpus is then used for training the contextual LM. This new contextual LM is used for further decoding.

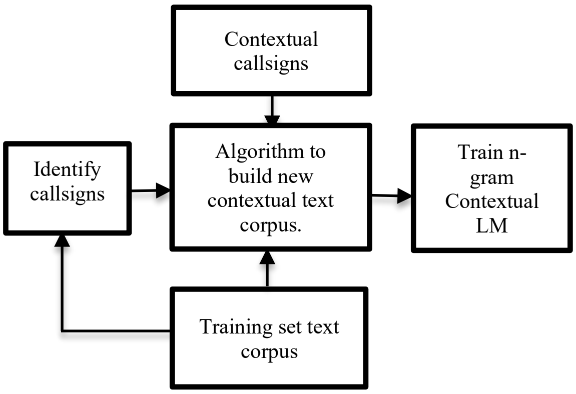
Contextual callsigns
Identify callsigns
Algorithm to build new contextual text corpus.
Train n-gram Contextual LM
Training set text corpus

Figure 5. A diagram illustrating the process of utilizing contextual callsigns to construct a contextual Language Model (LM).

B. *N-Best matching method*

While the contextual LM method can improve the CRR, it cannot be applied to real-time scenario. Building contextual LM and HCLG graph will take 10-15 minutes in a reasonably good CPU system, by which time the contextual information becomes irrelevant for the ASR to make any improvements in CRR. When an aircraft enters the airspace of a country, the ADS-B system located at the nearest ATC command center can receive its callsign among other information. By processing this information along with the flight plan details, ATM system can provide the callsign information to the CSR, albeit with a possible delay of few minutes. But since the aircraft is going to be in the same airspace for the next 10-15 minutes, CSR can make use of the contextual callsign information for improving its callsign recognition rate.

The algorithm we developed and shown in Figure 6 can process contextual callsign information in real-time, so there is no further delay in transcribing when compared to normal ASR engine. In this method, the lattice to n-best selects the n-best sequences, and each callsign is compared with the word sequence in the n-best list. Once a match is found, the whole word sequence is selected as the output. Otherwise, a 1-best is selected as the output as in the case of normal ASR

C. *Experimental Set up*

As described earlier, in a real system, contextual information can be obtained from surveillance or flight plan data. This contextual information is provided for a window of fixed duration, typically 1 hour. In the experiments we have used oracle contextual callsigns, which are obtained from transcribed test data by using our callsign detection script. In the deployment scenario, these callsigns can be obtained from a system that provides contextual callsigns that are relevant to a window of fixed duration of 1 hour.

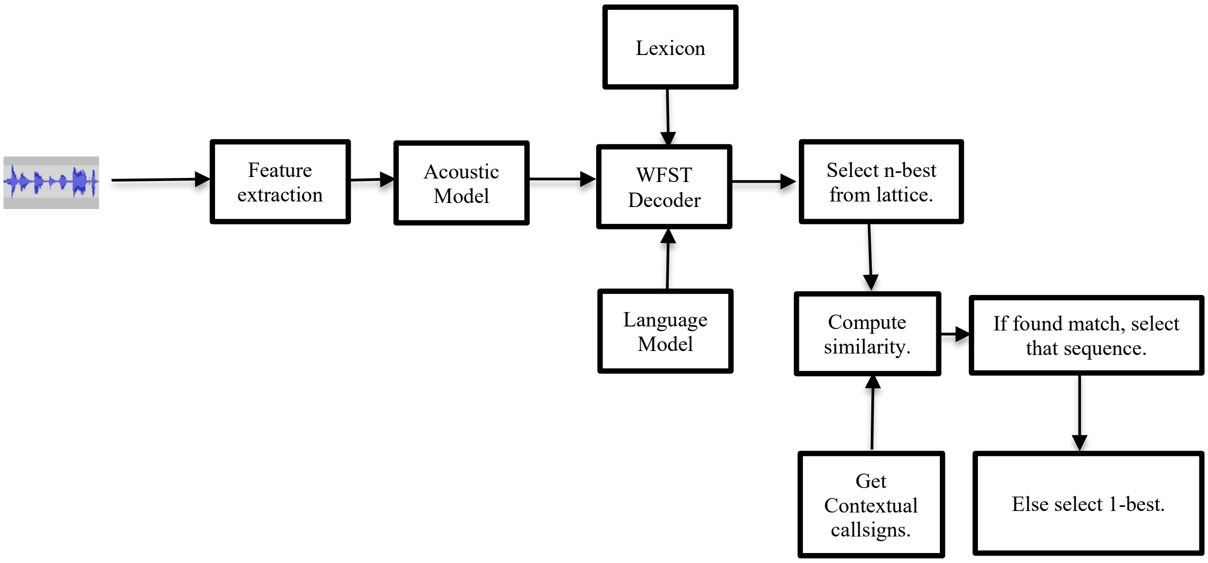

Figure 6. Major blocks in real-time Contextual Speech Recognition. (N-Best Method)

D. Results and Discussions

In this section we discuss CRR performance for both baseline ASR and CSR.

- *Contextual LM Method*

  For this method, the test set is a CAAS test set of 3.28 hours duration which is different from the test set described under section III-C results and discussions. So, for this method, the contextual callsigns are obtained from the transcripts by using our callsign detection algorithm.

  TABLE-II CRR RESULTS FROM CONTEXTUAL LM METHOD

Test Set Name	Number of unique callsigns	CRR (Baseline)	CRR (CSR)
test_C_APP	67	98.66	99.11
test_C_ENROUTE	49	97.39	99.35
test_C_TOWER	85	93.25	98.41
test_P_APP	81	93.59	98.58
test_P_ENROUTE	57	93.1	98.28
test_P_TOWER	116	93.27	96.77
Average		94.87*	98.4*

  The baseline ASR used for this experiment is a Kaldi Based HMM-DNN model from section III-B.

- *N-best Matching Method*
  This method can be applied to any n-best producing ASR. For the experiments, we have considered the Kaldi HMM-DNN baseline model mentioned in section III. Each test file is of 1hour duration and top 6 files are from channel 123.7MHz from 13th June 18th June,2023. While the last 6 files are from channel 133.8 MHz from 13th June to 18th June in that order.

TABLE-III CRR RESULTS FROM N-BEST MATCHING

Filename	Unique Callsigns	Baseline CRR	CRR
123.7MHz-13-06-2023	82	86.59	95.12
123.7MHz-14-06-2023	71	94.37	98.59
123.7MHz-15-06-2023	70	94.29	100.00
123.7MHz-16-06-2023	93	86.02	94.62
123.7MHz-17-06-2023	79	88.61	96.20
123.7MHz-18-06-2023	62	88.71	96.77
133.8MHz-13-06-2023	43	95.35	97.67
133.8MHz-14-06-2023	56	82.14	92.86
133.8MHz-15-06-2023	101	98.02	99.01
133.8MHz-16-06-2023	71	88.73	91.55
133.8MHz-17-06-2023	66	83.33	86.36
133.8MHz-18-06-2023	30	86.67	96.67
Average		**89.4**	**95.5**

We have chosen the Kaldi model as the baseline ASR for N-best matching experiments. Wenet based E2E also provides N-best output during decoding. From Tables II & III we see that the performance of the contextual LM method is better than N-best matching method. This is expected, as we are restricting the search space in the LM method, and hence the search algorithm should work better. In the ATC scenario, WER is not a comprehensive metric, but recognition performance of callsigns, commands, values etc. are important.

V. CONCLUSION AND FUTURE WORKS

We have presented a full pipeline of ASR for ATC focusing on the Singapore region air space. Two ASR systems from scratch were trained using the data from CAAS. The fine-tuned pre-trained model with LM has better WER performance than hybrid and E2E models. By making use of the contextual callsign information, we could improve the callsign recognition rate by 6.01%. The system is simple, and processing is real-time. In the future work, we plan to include various ATC commands apart from the callsigns in the contextual list.

ACKNOWLEDGMENT

This research / project is supported by the National Research Foundation, Singapore, and the Civil Aviation Authority of Singapore, under its Voice Communication Digitalization and Analysis for Air Traffic Management (ATM) project (Award No.: ATP_IOP for ATM_I2R_1)

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and the Civil Aviation Authority of Singapore

REFERENCES

[1] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek,N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz et al., “The Kaldi speech recognition toolkit,” in IEEE 2011 workshop on automatic speech recognition and understanding, no. CONF.IEEE Signal Processing Society, 2011.

[2] Z. Yao, D. Wu, X. Wang, B. Zhang, F. Yu, C. Yang, Z. Peng,X. Chen, L. Xie, and X. Lei, “Wenet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit,” in ISCA Conference of the International Speech Communication Association (Interspeech), 2021, pp. 4054 – 4058.

[3] J.M. Cordero, M. Dorado, and J.M. de Pablo, “Automated speech recognition in atc environment,” in Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems, 2012, pp. 46–53.

[4] H. Helmke, J. Rataj, T. Mühlhausen, O. Ohneiser, H. Ehr, M. Kleinert, Y. Oualil, M. Schulder, and D. Klakow, “Assistant based speech recognition for ATM applications,” in Proc. of 11th USA/Europe Air Traffic Management Research and Development Seminar (ATM 2015), Jun. 2015.

[5] Juan Zuluaga-Gomez , Petr Motlicek , Qingran Zhan , Karel Vesely , Rudolf Braun, “Automatic Speech Recognition Benchmark for Air-Traffic Communications,” in Proc. of Interspeech,2020, pp. 2297–2301.

[6] J. Zuluaga-Gomez, A. Prasad, I. Nigmatulina et al., “How does pre-trained wav2vec 2.0 perform on domain-shifted asr? an extensive benchmark on air traffic control communications,” in Proc.SLT, 2022.

[7] https://github.com/jlvdoorn/WhisperATC

[8] J. Zuluaga-Gomez, K. Vesel`y, I. Sz¨oke, P. Motlicek, M. Kocour,M. Rigault, K. Choukri, A. Prasad, S. S. Sarfjoo, I. Nigmatulinaet al., “ATCO2 corpus: A Large-Scale Dataset for Researchon Automatic Speech Recognition and Natural Language Understanding of Air Traffic Control Communications,” arXiv preprint arXiv:2211.04054, 2022.

[9] 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, no. 5, p. 490, May 2023,ISSN: 22264310. DOI: 10.3390/aerospace10050490. [Online]. Available: https://doi.org/10.3390/aerospace10050490.

[10] https://lindat.mff.cuni.cz/repository/xmlui/handle/11858/00-097C-0000-0001-CCA1-0

[11] https://www.liveatc.net/search/?icao=wsss

[12] Jzuluaga/wav2vec2-large-960h-lv60-self-en-atc-uwb-atcc-and-atcosim · Hugging Face

[13] M. Kocour, K. Vesel`y, A. Blatt, J. Z. Gomez, I. Sz¨oke, J. Cernocky, D. Klakow, and P. Motlicek, “Boosting of Contextual Information in ASR for Air-Traffic Call-Sign Recognition,” in Proc.Interspeech 2021, 2021, pp. 3301–3305.

[14] A. Blatt, M. Kocour, K.Veselý, I. Szöke, D.Klakow “Call-sign recognition and understanding for noisy air-traffic transcripts using surveillance information”, https://arxiv.org/abs/2204.06309

[15] I Nigmatulina, S Madikeri, E Villatoro-Tello, P Motliček, J Zuluaga-Gomez, K.Pandia, A.Ganapathiraju” Implementing contextual biasing in GPU decoder for online ASR” arXiv preprint arXiv:2306.15685

[16] Y. Oualil, M. Schulder, H. Helmke, A. Schmidt, and D. Klakow, “Real-time integration of dynamic context information for improving automatic speech recognition,” in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH,2015, vol. 2015-Janua, pp. 2107–2111

[17] J. Zuluaga-Gomez, S. S. Sarfjoo, A. Prasad, I. Nigmatulina, P. Motlicek, K. Ondrej, O. Ohneiser, and H. Helmke, “ BERTraffic:BERT-based joint speaker role and speaker change detection for air traffic control communications,” arXiv preprint arXiv:2110.05781, 2021.

[18] H. Helmke, O. Ohneiser, J. Buxbam, and C. Kern, “Increasing ATM Efficiency with Assistant Based Speech Recognition,” in Proc. of the 13th USA/Europe Air Traffic Management Research and Development Seminar, Seattle, USA, 2017.

[19] TL Nwe, H Sun, B Ma, H Li, “Speaker clustering and cluster purification methods for RT07 and RT09 evaluation meeting data’,’ IEEE transactions on audio, speech, and language processing 20 (2), 461-473

[20] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahrmani, Vimal Manohar, Xingyu Na, Yiming Wang and Sanjeev Khudanpur, “Purely sequence-trained neural networks for ASR based on lattice-free MMI” in Proc. Interspeech, 2016, pp.2751-2755

[21] Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohamadi, and Sanjeev Khudanpur, “ Semiorthogonal low-rank matrix factorization for deep neural networks,” INTERSPEECH, 2018, pp. 3743-3747

[22] Mehryar Mohri, “Weighted automata algorithms,” in Handbook of weighted automata, 2009.

[23] Tom Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, “A study on data augmentation of reverberant speech for robust speech recognition,” in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017, pp. 5220–5224

[24] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu,W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, “ Conformer:Convolution-augmented Transformer for Speech Recognition,” in Proc. Interspeech 2020, 2020, pp. 5036 – 5040.

[25] P. Rangan, S. Teki, and H. Misra, “Exploiting spectral augmentation for code-switched spoken language identification,” arXiv preprint arXiv:2010.07130, 2020.

[26] https://huggingface.co/facebook/wav2vec2-conformer-rel-pos-large-960h-ft