The HAAWAII Framework for Automatic Speech Understanding of Air Traffic Communication

Hartmut Helmke, Matthias Kleinert, Arthur Linß
Institute of Flight Guidance,
German Aerospace Center (DLR)
Braunschweig, Germany
hartmut.helmke@dlr.de; matthias.kleinert@dlr.de; arthur.linss@dlr.de

Petr Motlicek
Idiap Research Institute, Martigny, Switzerland
petr.motlicek@idiap.ch

Hanno Wiese
Fraport AG Frankfurt Airport Services Worldwide,
Frankfurt, Germany, h.wiese@fraport.de

Lucas Klamert
Austro Control, Vienna, Austria
lucas.klamert@austrocontrol.at

Julia Harfmann, Nuno Cebola
NATS, Whiteley, Fareham, United Kingdom
julia.harfmann@nats.co.uk; nuno.cebola@nats.co.uk

Hörður Arilíusson, Teodor Simiganoschi
Isavia ANS, Reykjavík, Iceland
hordur.ariliusson@isavia.is; teodor.simiganoschi@isavia.is

***Abstract*—During the last decade many successful applications combining Automatic Speech Recognition and Understanding (ASRU) for Air Traffic Management applications have been proposed and demonstrated. The HAAWAII project developed a generic architecture and framework, which was validated for, e.g., callsign highlighting, pre-filling radar labels and readback error detection. It supports recognizing and understanding pilot and air traffic controller (ATCo) transmissions. Contextual information extracted from available surveillance data, from flight plan data and from previous transmissions can be exploited to significantly improve ASRU performance. Different design decisions have been taken, depending on concrete scenarios. This paper evaluates the effect of the design decisions integrated in the HAAWAII framework on overall performance for speech understanding based on eight hypotheses, of which seven are validated. Using all framework elements enables command recognition rates for ATCos of 90% for real-time applications and 93% for offline applications, respectively. The most significant impact is achieved, when callsign information from surveillance data is available: the command recognition rate improves by more than 20% absolute. Knowing apriori, whether ATCo or pilot is speaking, can provide additional improvement in command recognition rate up to 16% absolute. The reported results are based on commands from apron, approach, and enroute recorded both in laboratory and in ops room environment.**

Keywords—Speech Recognition; ABSR; ASRU; Speech Understanding; HAAWAII framework; Air Traffic Control; Voice Recognition

I. INTRODUCTION

A. Problem

During the last decade many successful applications of Automatic Speech Recognition and Understanding (ASRU) for Air Traffic Management (ATM) have been demonstrated. Supporting Air Traffic Controllers (ATCos) by prefilling radar label entries with ASRU based on the architecture described in this paper has achieved a Technology Readiness Level (TRL) of 6, which was validated in SESAR 2020 funded industrial research [1].

Many ASRU applications in ATM require a real-time reaction. However, applications such as ATCo workload prediction based on the digitized transmission can run offline without caring about the real-time aspect. This could increase recognition performance as more computing time can be used to recognize the spoken transmission. ASRU needs Speech-to-Text (S2T), which transforms an analog or digital speech signal into a sequence words, whereas the Text-to-Concept (T2C) part is the understanding part. It transforms the sequence of words into ATC concepts like callsigns, command types, command values, and command conditions. As different requirements exist for different ASRU applications, also different architectures might be needed. For each application the following research questions need to be answered:

- How to decide in real-time, when the transmission starts and when it ends, especially when Push-To-Talk (PTT) information is not available?
- How to implement the Speech-to-Text transformation, e.g., (1) as continuous (real-time) process, (2) only as offline solution executed at the end of the communication, or (3) even as different engines specifically for pilot and ATCo?
- How to decide, whether ATCo or pilot speaks?
- How to implement Text-to-Concept extraction, e.g., (1) continuous semantic extraction, (2) semantic extraction only at the end of a transmission, or (3) different instances for pilot and ATCo semantic extraction?
- Is a special recognition engine for callsigns needed?
- How to implement plausibility checking? Should it be implemented at the end of the understanding process so that it only enables deleting extractions with low plausibility or during the command understanding process, which enables to re-interpret rejected extractions?

B. Suggested Solution

The HAAWAII (**H**ighly **A**utomatic **A**ir traffic controller working position **W**ith **A**rtificial **I**ntelligence **I**ntegration) project [2] has developed a generic framework, which answers the above questions.

Supported by SESAR Joint Undertaking, grant agreement No 884287 and 874464.

C. Paper Structure

Section II gives an overview of related work starting with ASRU applications and achievements in ATM and continuing with different architectures suggested for ATM applications. Sections III describes the HAAWAII framework concentrating on the Speech Understanding part. Section IV describes the validation setup, which includes validation hypotheses, metrics, and description of the used data sets. Section V reports on the validation results before the final conclusions in section VI.

II. Related Work

A. Related Work for Speech Recognition and Understanding

Over the last 70 years, advances have led to dramatic improvements in the field of Automatic Speech Recognition (ASR). An overview of the first four decades is provided by, e.g., Juang and Rabiner [3]. Connolly from FAA [4] was one of the first to describe the steps of using ASR in the ATM domain. In the late 1980s, a first approach to incorporate speech technologies in ATC training was reported [5] to replace expensive simulation pilots.

The challenges with ASR in ATC today go beyond basic training scenarios, where often ICAO phraseology [6] is followed very closely. Modern ASR applications have to recognize experienced controllers with various accents, who more often make deviations from the mentioned standards. Nowadays, ASR is for example used to obtain more objective feedback concerning controllers' workload [7] or readback error detection in the US [8] or in Europe [9]. A good overview of the integration of ASR in ATC is provided in the paper of Nguyen and Holone [10]. A more technical overview is given by Lin [11].

Radar Label Maintenance supported by Automatic Speech Recognition and Understanding (ASRU) has recently achieved a Technological Readiness Level (TRL) of 6 being validated in DLR's ATMOS simulation environment [12]. This development has started in 2013, when it was shown that Speech Recognition and an Arrival Manager in combination improve each other [13]. More systematically, this was analyzed in 2015. The term Assistant Based Speech Recognition (ABSR) was born [14]. It was shown that application of ABSR reduces ATCos' workload [15]. The same validation trials of the AcListant®-Strips project also showed that ABSR reduces fuel burn by 60 liters of kerosene per arrival [16]. The MALORCA project showed how to automatically adapt ABSR to different approach areas, i.e., Vienna and Prague, by means of machine learning [17]. Commercial-of-the-shelf ASR engines were unable to achieve the performance of the MALORCA approach, at least in 2019 [18].

Since speech recognition does not include speech understanding, European ATM partners agreed on a so-called ontology to ease understanding of approach controller utterances [19] being extended to apron controller utterances in the STARFiSH project [20] and even more important to pilot transmissions [21]. Ontologies for speech understanding were not only evaluated and implemented in Europe. Chen et al. compare the European and US ontologies [22] [23]. The term ABSR was extended to ASRU as it is already common practice in the *normal* speech recognition community for a long time. The experiments of the AcListant®-Strips project for Dusseldorf approach were repeated for Vienna approach in 2022 with 12 ATCos from Austro Control in the context of ASRU. ATCos' clicking time could be reduced from 12,700 seconds down to roughly 400 seconds, a factor of 31 [1]. The clicking time is the time between opening a menu for e.g. selecting a flight level value until clicking on the selected value, i.e. in the example the selected flight level. Only 4% of the given ATCo commands were missing or wrong in the radar label cells when supported by ASRU. Current operational practice in the ops room is manual command input without ASRU support. In this case 11% missing or wrong inputs were observed [12]. The deployed architecture was developed in the HAAWAII project and is described in the next section. Before we give a short overview of other architectures to support ASRU in ATM.

B. Related Work with Respect to Architectures

In the context of ABSR, DLR and Saarland University have introduced the architecture, which is based on an available Arrival Manager [14], shown in the left part of Figure 1.

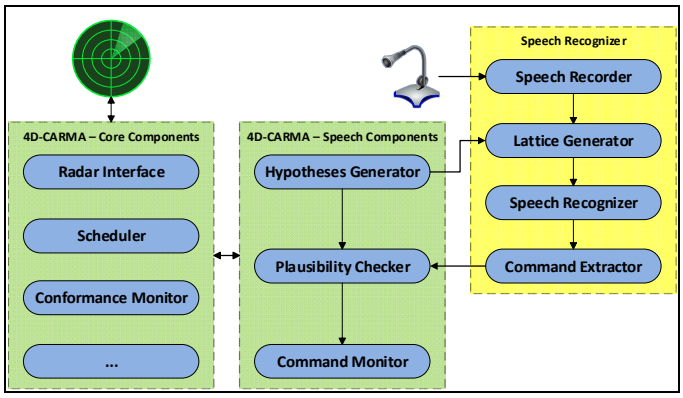

Figure 1. Assistant-Based Speech Recognition Architecture based on an Arrival Manager

The middle part of Figure 1 shows the modules relevant for this paper. The *Hypotheses Generator* creates hypotheses about possible ATCo commands, which includes callsign, command types and also command values. The *Plausibility Checker* checks whether extracted commands make sense in the current situation. The *Command Monitor* continuously checks, whether previously recognized command are consistent with radar data.

22 European ATM partners have agreed on the building blocks of ASR in the EATMA architecture (European Air Traffic Management Architecture) [18] as shown in Figure 2.

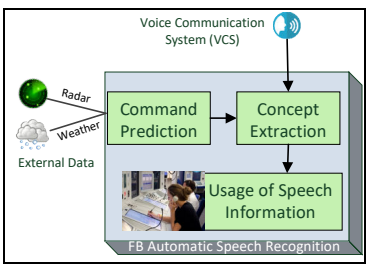

Figure 2. Integration of Automatic Speech Recognition into EATMA (taken from [18])

The functional block *Automatic Speech Recognition* receives an audio signal as input and transforms it into a sequence of words, which is transformed into a sequence of ATC concepts by the *Concept Extraction* module. The resulting concepts can be used for further applications, i.e. by *Usage of Speech Information*. The *Command Prediction* function is optional predicting full commands or just callsigns. More details are described in [24]. The EATMA architecture of ASR was improved in Solution PJ.10-W2-96 ASR of the SESAR 2020 Industrial Research in 2023; see Figure 3 taken from [25]. The *Command Prediction* function is kept. The *Concept Extraction* function of Figure 2 is split into *Recognize Voice Words*, also known as *Speech-to-Text,* and *Apply Ontology and Logical Checks*, which corresponds to *Text-to-Concept* in the HAAWAII framework.

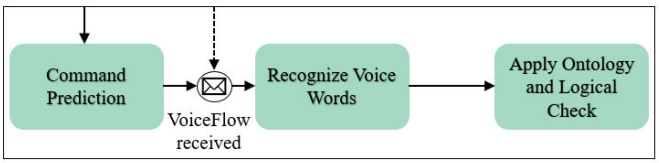

Figure 3. Resource orchestration view: Differentiation between Speech Recognition (middle part) and Speech Understanding (right side) [25]

Figure 4 shows the ASRU pipeline suggested by MITRE [26]. It addresses detecting start and end points and the speaker classification, i.e., whether ATCo or pilot is speaking.

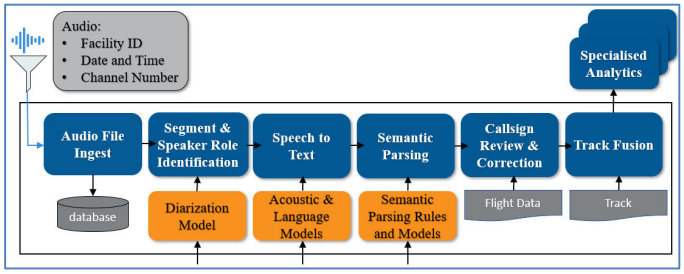

Figure 4. Recognition and Understanding pipeline suggested by MITRE [26]

III. THE HAAWAII FRAMEWORK

The HAAWAII framework describes the process of how to transform an ATC audio signal into conceptual elements, which can be incorporated into all sorts of ATM applications. We first describe the full framework for automatic speech recognition and for automatic speech understanding. Then detail the speech understanding framework, before we describe the final consistency check of the extracted commands in the last subsection

A. *Automatic Recognition and Speech Understanding Framework*

The ASRU framework considers ATCo and pilot voice transmissions. It provides means to link both together to benefit from the dialogue nature of ATCo-pilot communications. The core components of the HAAWAII framework are shown in Figure 5. The blue arrows show the way of the audio signal or more precisely the information derived from the audio signal by the different processing steps.

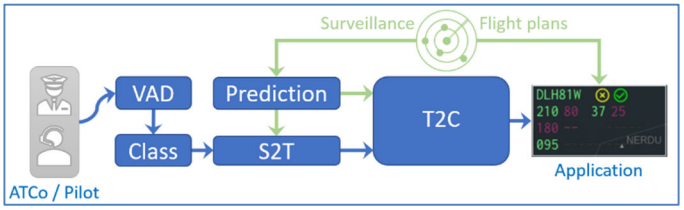

Figure 5. The HAAWAII Framework

The green arrows show how surveillance data and flight plan information are incorporated into the process to enhance the overall quality. The major input into the HAAWAII framework is an audio signal containing ATCo-pilot communications. It considers that the audio signal might not come in the form of already split transmissions, but it is prepared to accommodate the fact that all transmissions might be within a single audio stream. Therefore, the framework implements **Voice Activity Detection** (*VAD*), which splits the continuous audio stream into individual transmissions. A *VAD* is prone to errors. Either we can have over-splittings, i.e. splitting within a transmission or under-splittings, i.e. no splitting between independent transmissions. Splitting too late in a real-time application also implies a late final output to the application. When *VAD* indicates the start of a transmission the audio signal is immediately forwarded to the so-called **speaker classification** (*Class*) deciding, e.g., whether the ATCo or the pilot is speaking.

After *Class* the audio signal is passed into a **Speech-to-Text** (*S2T*) component to initiate the recognition process on a text-based level, i.e. the audio signal is transformed into a sequence of words. The framework supports that the used *S2T* component is capable of producing intermediate recognitions, i.e., *S2T* does not need a complete voice transmission to produce reasonable outputs. Instead, it can continuously receive audio and updates the recognized sequence of words, until the end of a voice transmission is reached, i.e., *S2T* could provide word sequences in increments, which allows analysis of a voice transmission even before it has ended.

The HAAWAII framework overall aims at producing a high-quality output for speech-based ATM applications. For this purpose, it integrates contextual knowledge wherever possible without the need for many additional sources of information. Contextual information is incorporated by the integration of surveillance data and flight plan information. The framework describes for that purpose a component to **predict callsigns and/or commands** (*Prediction*), which are likely to be part of a voice transmission in the near future. This information can then be forwarded to *S2T* and *T2C* to improve the recognition performance. For example, a callsign abbreviated on purpose in the voice communication can only be recognized correctly, if the full form of the callsign is available from contextual knowledge.

B. *Speech Understanding Framework*

The output of *Prediction* and *S2T*, regardless of whether it is an intermediate increment or the final transmission, is forwarded to the **Text-to-Concept** (*T2C*) component often also referred to as Concept Recognition. *T2C* performs a semantic interpretation (speech understanding) of the word sequence content. The result is a transformation to instructions with conceptual elements such as callsign, type, value, unit, qualifier etc. as defined in the ontology for ATC communication in [19].

This incremental output from *S2T* and subsequently also from *T2C* allows the application to benefit very early from the callsign information if the callsign is said in the beginning of the transmission. Also, some instructions are available before the complete transmission is finished. The following paragraphs describe in more detail, how the different blocks of *T2C* shown in Figure 6 interact with each other to produce the described output. Whenever an output from *S2T* reaches the *T2C* component it is used as input into the *Understand* block, which has three different instances, one if the speaker was detected as pilot, one for ATCo and a third, when no speaker information is available. With the information from *Class* this block selects the appropriate instance and applies the mentioned ontology to make the transformation into commands (instructions).

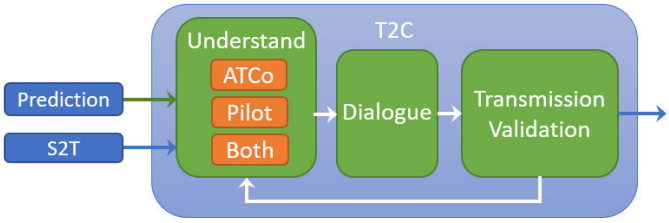

Figure 6. T2C including understanding and context from dialogue

The *Understand* block also includes a type-dependent validation of each individually extracted command, e.g., parts of the transmission may indicate a QNH command by the maybe wrongly recognized word *QNH*, but no four-digit related QNH-value can be found or a possible frequency change is detected, but the recognized frequency is not among the allowed frequencies configured for a given sector or area. If such a validation fails, all words from the word sequence that were linked within the process to be relevant for the respective command remain free to be possibly linked with other commands.

C. Consistency Check of Understanding

The *Dialogue* block following the *Understand* block tries to eliminate ambiguities by incorporating information from the pilot-ATCo dialogue. Especially pilots often abbreviate readbacks, because they directly respond to clearances from an ATCo. A pilot response could perhaps just be "one hundred roger". In this example, only contextual dialogue knowledge from the preceding ATCo transmission makes it possible to figure out, which aircraft did respond and if "one hundred" was a response to e.g. an instructed heading or a flight level.

When the whole ATCo-pilot communication has been recognized, the final validation of all callsigns, commands etc. are validated by *Transmission Validation*. This ensures that not both a turn to the RIGHT and LEFT direction are extracted from the same transmission, which is unlikely to be correct. Based on heuristics this block then removes all commands which are less probable.

The feedback loop in Figure 6 from *Transmission Validation* to *Understand* enables to correct errors made by the *VAD* block. For example, it might occur that *VAD* has decided to split the transmission "lufthansa one alfa taxi november november eight" after the word "taxi", because the ATCo made a short pause, and *VAD* detected this as end of transmission. This error by *VAD* means that *T2C* receives two independent transmissions "lufthansa one alfa taxi" and "november november eight". Before sending the result of a transmission to the application, *T2C,* therefore, evaluates if two transmission, which appeared very close together, could be part of an error made by *VAD*. If that could be the case, the two transmission are combined into one and the whole process starting from *Understand* will be executed again.

First implementations of the framework already exist for TRL6 to support approach controllers for pre-filling radar label entries [1], for ATCo and pilot communication for London Terminal Maneuvering Area (TMA) and Isavia enroute airspace [9], to support Frankfurt apron controllers and simulation pilots [27], and to support multiple remote tower operations [28].

IV. Validation Setup

The first subsection enumerates the validation hypotheses. Subsection IV.B describes the metrics to verify or falsify the hypotheses and the last subsection IV.C presents the voice and surveillance data sets from Frankfurt apron, Vienna approach, London approach and TMA and Isavia's oceanic traffic, which were available to calculate the metrics.

A. Validation Hypotheses

S2T of HAAWAII is evaluated in [29]. We concentrate on the T2C and the *Prediction* components, shown in Figure 5. The following validation hypotheses based on the research questions from the introduction are evaluated:

H1. Integration of contextual knowledge, i.e. the list of available callsigns in the current airspace situation, into Concept Extraction, improves the command extraction performance.
H2. Integration of contextual knowledge improves the callsign extraction performance.
H3. Integration of contextual knowledge from the pilot-ATCo conversation, i.e. the previous utterances, improves callsign extraction performance and command extraction performance.
H4. Different models for pilot and ATCo command extraction outperform having just one common model.
H5. Integration of command validation into the *T2C* does not only reduce the command error rates, but also increases the command extraction rates.
H6. The *T2C* block can repair over-splittings of the Voice Activity Detection (VAD).
H7. Integration of plausibility values from *S2T* on word level improves *T2C* performance.
H8. Integration of plausibility values from semantic interpretations improves *T2C* performance.

B. Metrics

For validating or falsifying the hypotheses of the last subsection we use the following metrics, which are introduced in [14] and detailed by Chen et al. [23], resulting in a simple scheme for measuring performance on semantic level. The scheme is independent of semantic concept type or subcomponents and treats all semantic components with equal importance.

TABLE I. DEFINITION OF BASIC METRIC ELEMENTS

Name	Definition
TP: True Positive	Total number of True Positives: The concept is present and correctly and fully (including all subcomponents) detected.
FP: False Positive	Total number of False Positives: The concept is incorrectly detected, i.e., either the concept is not present at all or one or more of its subcomponents are incorrectly detected.
TN: True Negative	Total number of True Negatives: The concept is correctly not detected, because the concept is not present.
FN: False Negative	Total number of False Negatives: A concept is not detected when it should have been.
TA: Total	Total number of annotated concepts, i.e., gold concepts.

We use the metric for both the performance for command extraction and for callsign extraction. Table I lists definitions that are the building blocks for the performance metrics. From the five building blocks we can derive *recognition rate* and *recognition error rate* (Eq.1, 2). Additionally, we define in Eq.4 the *Fα-Scores* by defining *Recall* and *Precision* (Eq.3), where α is here a parameter to either emphasize precision or recall.

$$RcR = Recognition\ Rate = \frac{TP + TN}{TA} \quad (1)$$

$$RER = Recognition\ Error\ Rate = \frac{FP}{TA} \quad (2)$$

$$Recall = \frac{TP}{TP + FN}; Precision = \frac{TP}{TP + FP} \quad (3)$$

$$\mathrm{F}_{\alpha}\mathrm{Score} = \frac{(1 + \alpha^2) * \mathrm{Recall} * \mathrm{Precision}}{(\alpha^2 * Precision) + Recall} \quad (4)$$

C. Description of Available Data Sets

Overall, six different data sets, described in table II, are used for the evaluation of the hypotheses. Two data sets result from simulation environment (lab) and four from the operational environment. Noisy pilot recordings and ATCo transmissions are available. Row *"#Transmissions*" contains the number of different transmissions consisting of one or up to eight different commands, whose sum is shown in row "*#Cmds*". Transmissions are only considered, if both the manual transcription and the manual annotation, i.e. the extracted commands, are available. Row "*WER*" shows word error rates calculated as Levenshtein distances between reference transcript and hypothesized ASR output. WER varies between 1.8% and 6.2% for different dataset and presents the quality of the *S2T* output, which is the base input for *T2C* and, therefore, influences its performance. Rows "*ReR*" show the achieved command recognition rates and callsign recognition rate (*CsgnR*) for an ideal S2T engine with a WER of 0%. The performance is worse for the data from the ops room. Row "No class" shows the percentage of words, which were not used by the command extraction algorithm to extract the commands. The enroute transmissions from Isavia show a high variability in the used word sequence, which often do not contribute to a command.

Row "Ø CsgnP" shows the average number of callsigns, that the *Prediction* module assumes will receive a command in the next few minutes. The number is quite high for the NATS airspace with overflights, departures, arrivals and VFR flights. Row "Csg Pred Err" shows the callsign prediction error, i.e. the percentage of callsigns, which get a command, but are not in the set of predicted callsigns.

TABLE II. DATA SETS FOR HYPOTHESES EVALUATION

	Frankfurt	Vienna App	NATS ATCo	NATS Pilot	Isavia ATCo	Isavia Pilot
Noise cond.	Lab	Lab	Ops room		Ops room	
Flight Phase	Apron	Appr	London TMA		Enroute	
Year	2022	2022	2020		2020	
#Transmissions	6362	8856	2060	2415	1484	1765
# Cmds	15495	17096	3596	4404	3012	3465
WER	3.2%	3.1%	1.8%	4.6%	3.1%	6.2%
RcR	97.1%	99.1%	92.3%	91.0%	92.2%	89.2%
CsgnR	99.1%	99.8%	98.4%	98.2%	98.1%	97.8%
No class	6.0%	2.8%	10.2%	10.2%	20.6%	19.4%
Ø CsgnP	9	21	51	51	19	19
Csg Pred Err	2.3%	0.02%	4.1%		5.5%	
N of t-Test	55	48	19	18	13	13

In the lab environment the number is quite small. No flight plan information, but only surveillance data, was available for Isavia's airspace with the consequence that all callsign with designator FEI (arctic eagle) were not predicted, which explains the high number of spoken, but not predicted callsigns. Row "*N of t-Test*" contains the number of different speaker sessions lasting from 30 to 120 minutes being available for the performed t-tests to check for statistical significance in the following section with the results.

V. VALIDATIONS RESULTS

The results presented in this section follow the structure shown in Table III. The main columns always differentiate between performance on "Command" and "Callsign" level. "Command" refers to the performance on complete commands including all relevant elements such as callsign, type, unit, qualifier etc. "Callsign" solely refers to the correct or wrong recognition of the aircraft callsign. The results in these columns are separated in the child columns "Recognition" and "Error", which include the achieved metric value (Rate) and the p-value (α) gained by paired t-tests to show if the results with respect to the presented hypotheses are statistically significant or not. We defined p-values below 5% as statistically significant and mark cells in green, if that is achieved.

We use different color coding taken from [1] to visualize, how statistically significant a hypothesis is: Light green color is used for 5%<= p-value <10%. Negative values point to evidence that the counter hypothesis could be true. Orange color shows that a counter hypothesis got a p-value between 0% and -5%. If the counter hypothesis would have been only slightly supported, i.e., we have p-value between -5% and -10%, we would have used light red. This case, however, does not occur in our data set. In cases, with statistical evidence neither for the hypothesis nor for the counter hypothesis, we use a yellow color. In summary, positive p-values indicate a trend for the hypothesis and negative values for the counter hypothesis.

A. Integration of Surveillance or Flight Plan Data

First, we verify the hypotheses H1 and H2, whether using callsign information from surveillance or flight plan data improves the performance of T2C. In baseline scenarios (rows with "B" in Table III) the T2C component received information about available callsigns.

TABLE III. CONTEXT INFORMATION INCREASES PERMORMANCE

H1, H2		Command				Callsign			
		Recognition		Error		Recognition		Error	
		Rate	α	Rate	α	Rate	α	Rate	α
Frankf Apron	B	91.9%	0.0%	3.3%	0.0%	97.1%	0.0%	1.5%	0.0%
	H	76.2%		10.7%		81.1%		9.8%	
Vienna App	B	92.8%	0.0%	2.7%	0.0%	97.7%	0.0%	1.0%	0.0%
	H	66.7%		22.1%		70.8%		21.5%	
NATS ATCo	B	91.1%	3.E-06	2.9%	6.E-05	97.6%	7.E-06	1.4%	9.E-04
	H	80.3%		7.6%		86.2%		5.7%	
NATS Pilot	B	84.8%	7.E-06	6.7%	9.E-06	96.5%	1.E-05	2.7%	1.E-05
	H	69.3%		13.8%		78.4%		11.1%	
Isavia ATCo	B	90.1%	0.0%	4.9%	0.0%	96.9%	0.0%	1.6%	0.1%
	H	81.2%		7.0%		86.5%		4.4%	
Isavia Pilot	B	81.3%	0.0%	5.7%	0.0%	93.0%	0.0%	2.7%	0.0%
	H	67.8%		8.4%		76.4%		7.5%	

In test scenarios to evaluate the hypotheses, the callsign information was not available (rows with "H"). The results in table III clearly show that using callsign information dramatically improves the extraction rates and reduces the error rates for all considered data sets for both ATCo and pilot utterances. The p-values of 0.0% mean that p-value is even less than 10^{-20}=1.E-20. The p-values for NATS and Isavia data are statistically very significant, but higher. The difference to Frankfurt and Vienna is the number of compared data sets. Isavia data consists of only 13 data sets, where Frankfurt data considers 55 data sets. Hypotheses H1 and H2 have been clearly validated.

B. *Integration of Context from last Utterance*

The next hypothesis verifies, whether it has benefits to exploit context information from the last transmission, when interpretation of the current transmission is done (block *Dialogue* of *T2C*). Table IV shows the results. "*undef*" means, that statistical significance cannot be calculated, because the results for each data item are the same, i.e. the standard deviation is zero.

The yellow color in many cells indicates that using information from the last transmission has no real effect in those cases. Desired effects can be seen for interpretation of pilot utterances for both Isavia and NATS airspace and also for both command recognition and callsign recognition error rate. Callsign improvements are only seen for NATS. It is not surprising that we have no improvement for Frankfurt and Vienna data, because these data sets only contain ATCo transmissions. The error rate even goes statistically significant into the "wrong" direction, which hints to an error in the implementation as it obviously also links ATCo transmissions to each other. One example is a transmission in which the ATCo says "*lufthansa four **four** six*"; the *S2T* instead recognizes "*lufthansa four **five** six …*". The *Prediction* component tells us that *DLH4**4**6* and *DLH45**7*** are present. Both callsigns are equally close to the recognized words with just one deviation. The previous transmission five seconds ago was recognized as *DLH457*. Therefore, the *Dialogue* component assumes that this time it must also be *DLH457*. The hypothesis H3 is falsified, to improve ATCo transmissions if only ATCo and no pilot data is available.

TABLE IV. CONTEXT INFORMATION FROM LAST UTTERANCE

H3		Command				Callsign			
		Recognition		Error		Recognition		Error	
		Rate	α	Rate	α	Rate	α	Rate	α
Frankf Apron	B	91.9%	undef	3.3%	-0.9%	97.1%	undef	1.5%	-0.3%
	H	91.9%		3.2%		97.1%		1.4%	
Vienna App	B	92.8%	8.9%	2.7%	-1.5%	97.7%	7.8%	1.0%	-1.6%
	H	92.7%		2.1%		97.7%		0.5%	
NATS ATCo	B	91.1%	undef	2.9%	16%	97.6%	undef	1.4%	undef
	H	91.1%		3.0%		97.6%		5.7%	
NATS Pilot	B	84.8%	3.E-04	6.7%	-5.E-03	96.5%	3.8%	2.7%	undef
	H	83.9%		6.5%		96.3%		2.7%	
Isavia ATCo	B	90.1%	undef	4.9%	undef	96.9%	undef	1.6%	undef
	H	90.1%		4.9%		96.9%		1.6%	
Isavia Pilot	B	81.3%	0.3%	5.7%	8.5%	93.0%	32.0%	2.7%	-15.9%
	H	80.7%		5.8%		92.9%		2.6%	

C. *Different Models for ATCos and Pilots Transmissions*

For verifying the hypothesis on different models for ATCo and pilot transmissions, we have first evaluated the extraction performance with the generic model (rows "*Gen*" in table V), i.e., if no speaker information is available. Then it was analyzed what happens if the speaker's information is incorrect by using the pilot's model for the ATCo and vice versa (rows "*Wrong*").

TABLE V. USING GENERIC OR WRONG EXTRACTION MODEL

H4		Command				Callsign			
		Recognition		Error		Recognition		Error	
		Rate	α	Rate	α	Rate	α	Rate	α
Frankf Apron	Base	91.9%		3.3%		97.1%		1.5%	
	Gen	91.1%	3.E-08	4.0%	3.E-08	97.1%	undef	1.5%	undef
	Wrong	90.8%	9.E-10	4.2%	1.E-08	97.0%	18%	1.6%	12%
Vienna App	Base	92.7%		2.2%		97.7%		0.6%	
	Gen	92.7%	undef	2.2%	undef	97.7%	undef	0.6%	undef
	Wrong	90.4%	1.E-08	2.2%	41%	95.7%	3.E-07	0.6%	16%
NATS ATCo	Base	91.1%		2.9%		97.6%		1.4%	
	Gen	91.2%	-21%	3.4%	0.8%	97.5%	27%	1.7%	4.4%
	Wrong	91.1%	-44%	3.4%	0.6%	97.5%	23%	1.7%	5.7%
NATS Pilot	Base	84.8%		6.7%		96.5%		2.7%	
	Gen	81.6%	6.E-05	6.6%	39.8%	95.3%	0.2%	3.4%	3.4%
	Wrong	76.2%	5.E-09	7.3%	2.9%	95.2%	1.E-03	3.5%	1.5%
Isavia ATCo	Base	90.1%		4.9%		96.9%		1.6%	
	Gen	90.1%	15.9%	5.4%	1.E-03	96.9%	undef	1.6%	undef
	Wrong	74.4%	1.E-08	7.2%	5.E-04	96.7%	5.5%	1.8%	5.3%
Isavia Pilot	Base	81.3%		5.7%		93.0%		2.7%	
	Gen	78.7%	3.E-06	5.5%	-4.0%	91.9%	4.2%	3.1%	1.4%
	Wrong	68.9%	2.E-07	6.1%	31%	91.7%	2.9%	3.3%	5.E-03

The Frankfurt data shows only minor effects, but statistically significant for command extraction. The decrease in command recognition rate by 1.1% and 2.3% absolute is important for Frankfurt and Vienna, respectively, when using the wrong speaker model. No effects are observed for NATS ATCo data,

but for Isavia ATCos, the recognition rate improves compared to using the wrong model.

For pilot data, however, a big improvement for both NATS and Isavia data and for both command and callsign extraction is observed. Only the command recognition error rate for Isavia pilot data has changed into an unexpected direction. It decreases from 5.7% to 5.5%, when the generic model instead of the correct pilot model is used. The reason is on command level and not the wrong extraction of the callsign, because callsign recognition error rate increases when only using the generic model. Further investigation is needed, but on the other hand this is the prize for the better callsign extraction rate, which improves from 78.7% to 81.3. The F1-score, combining error rate and extraction rate, improves from 86.8% to 88.5, when knowing who is speaking. Hypothesis H4 has been validated.

D. *Intermediate and Final Semantic Command Checking*

Section III describes two validation (checking) steps included in the *T2C* component. The first validation is part of the *Understand* block and evaluates commands independent from each other. The second validation is performed at the end by *Transmission Validation*, validating commands in relation to each other. For hypothesis H5 we evaluate the influence of both validation steps separately. The results are shown in table VI. The "*Base*" rows contain the results, when all implemented validation steps are active. "*NoCheck*" means the individual validation of commands from the *Understand* block is switched off. The rows "*NoPost*" contain the results, when the post validation from *Transmission Validation* is switched off, where commands are validated in context to each other.

The results in table VI show that the individual validations of commands (rows "*NoCheck*") not only reduce the error rate, but also increase the command recognition rate.

TABLE VI. EFFECT OF EARLY AND LATE COMMAND CHECKING

H5		Command				Callsign			
		Recognition		Error		Recognition		Error	
		Rate	α	Rate	α	Rate	α	Rate	α
Frankf Apron	Base	91.9%		3.3%		97.1%		1.5%	
	NoCheck	91.8%	2.4%	3.3%	4.0%	97.1%	undef	1.5%	undef
	NoPost	91.8%	25%	3.4%	2.E-03	97.1%	undef	1.5%	undef
Vienna App	Base	92.7%		2.2%		97.7%		0.6%	
	NoCheck	92.6%	1.7%	2.4%	41%	97.6%	3.E-07	0.5%	-2.2%
	NoPost	91.4%	5.E-08	3.6%	2.E-04	97.7%	2.0%	0.6%	undef
NATS ATCo	Base	91.1%		2.9%		97.6%		1.4%	
	NoCheck	90.4%	5.E-03	3.3%	4.0%	97.1%	1.4%	1.4%	undef
	NoPost	91.1%	-33%	3.0%	16%	97.6%	undef	1.4%	undef
NATS Pilot	Base	84.8%		6.7%		96.5%		2.7%	
	NoCheck	83.6%	7.E-03	7.2%	6.E-04	95.4%	2.5%	2.5%	-2.7%
	NoPost	84.1%	7.E-04	7.5%	2.E-04	96.5%	undef	2.7%	undef
Isavia ATCo	Base	90.1%		4.9%		96.9%		1.6%	
	NoCheck	89.6%	0.2%	5.6%	3.E-03	96.9%	undef	1.6%	undef
	NoPost	89.2%	1.5%	5.8%	1.7%	96.9%	undef	1.6%	undef
Issavia Pilot	Base	81.3%		5.7%		93.0%		2.7%	
	Gen	81.1%	5.5%	6.2%	0.5%	92.9%	11%	2.8%	16%
	Wrong	80.3%	2.E-04	6.6%	7.E-04	93.0%	undef	2.7%	undef

The results are statistically significant for all data sets. We observed the highest improvement for NATS pilot data. The individual validation improves the command recognition rate from 83.6% to 84.8% and the error rate is reduced from 7.2% to 6.7%. The small differences between Base on the one hand and NoCheck and NoPost rows on the other hand show for some data sets that slight improvements are still possible. The callsign extraction performance is only slightly affected, because the checkings focus on the command types. Hypothesis H5 has been validated

E. *Repairing Over-Splittings from Voice Activity Detection*

For validating hypothesis H6, we used the data of Frankfurt Apron. Over-Splitting corrections was active during four of the five simulation days. As shown in table VII, 8849 utterances are considered. These are more than the 6362 utterances considered in table II, which only include utterances for which also manual annotations, i.e. speech understandings, are available.

TABLE VII. CORRECTION OF OVER-SPLITTING RESULTS

	Total	TP	FP	FN	Precision	Recall	F1-Score
Frankfurt Apron	8849	219	0	40	100.0%	84.6%	91.6%

We merge two utterances, i.e. assume over-splitting, if the time difference between the start of the second utterance and the end of the last utterance is small, i.e. <2.5 seconds, and we do not extract two different callsigns from them. The algorithm successfully merged 219 times. 40 times an over-splitting occurred, but it was not repaired. A merge never occurs, when no over-splitting has happened (FP=0). In 2.9% of the utterances we had an over-splitting, which could be repaired in 84.6% of the cases (column "*Recall*"). Most of 40 FN are observed, when the wrong over-splitting occurred within the callsign. All in all, repairing over-splitting improves recognition performance, i.e. the command recognition rate improves by 2.5%. We validated hypothesis H6.

The repairing algorithm is only the second-best choice. In an operational scenario a direct access to the push-to-talk (PTT) signal should be incorporated in the *VAD* process to avoid splitting problems at least for the ATCo transmissions.

F. *Improving Extraction Performance by Plausibilty Values*

Both *S2T* and *T2C* have plausibility values as output. *S2T* outputs so called N-best lists, i.e. we do not only get one sequence of words, but for each word the most probable ones together with plausibility values between 0.0 and 1.0. T2C also outputs a plausibility value for each extracted concept element. Accepting only extracted commands above a given threshold enables to find a compromise between a high command recognition rate RcR and a low command recognition error rate RER. Eq. (4) combines them both in the F-score. The participating air traffic controllers mostly put emphasis on low error rates. Therefore, we choose F-0.5, which puts more emphasis on the precision and not on recall. The precision considers the false positives, i.e. the errors. Figure 7 shows the F-0.5 scores for different plausibility values on command level. A plausibility value between 40% and 60% is a good compromise for high recognition and low error rates.

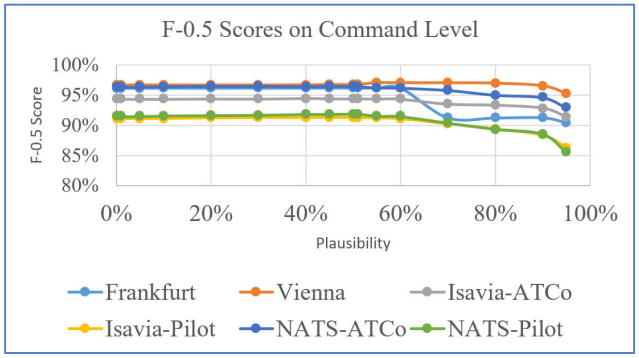

Figure 7. F-0.5 Scores on command level considering both word and semantic level

Figure 8 shows the F-0.5 scores for different plausibility values on callsign level. A plausibility value between 70% and 80% seems to be the best compromise. For Frankfurt we even got the best values for 95%.

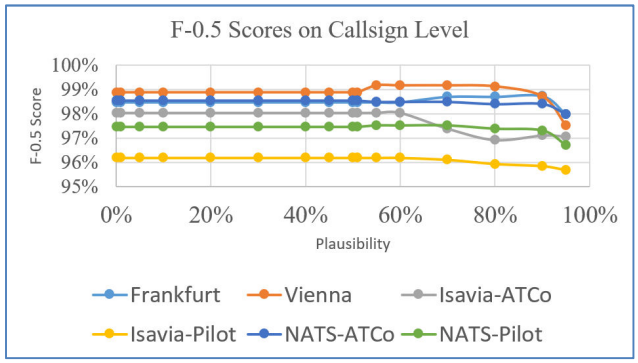

Figure 8. F-0.5 Scores on callsign level considering both word and semantic level

We calculated the F-0.5 scores considering plausibilities (1) only on word level, (2) only on semantic level, and (3) combining both levels. Considering only the word level is slightly worse. The hypotheses H7 and H8 are validated given the F-0.5 scores.

VI. Conclusions

A framework for Automatic Speech Recognition and Understanding (ASRU) for Air Traffic Management Applications was presented, which strictly distinguishes between transforming a speech signal into a sequence of words and its semantic interpretation. The framework was initiated by the HAAWAII project and successfully extended to industrial research with application for apron, remote tower, approach, sector, and enroute control. Our ASRU architecture enables a variety of air traffic control applications for all flight phases and has proven to be usable with lab and operational data. Some mechanisms such as integrating contextual knowledge have been clearly validated and are a *must* for all ASRU applications in ATM, whereas others might require tweaking or are dependent on the environment.

The framework enables to correct 85% of over-splitting errors resulting from voice activity detection enabling fast real-time and continuous recognitions without needing to wait for start of recognition, until the speaker has ended. Integration of context knowledge, i.e. callsign information from surveillance data and knowing whether ATCo or pilots talk improves command extraction rates by up to 30% absolute. Integration of callsign information is quite robust, i.e. missing callsigns and also long lists of more than 100 possible callsigns are tolerable.

Using context information from the last transmissions improves understanding of pilot utterances – who often abbreviate their readbacks – with positive effects on the recognition rates, but with small negative effects on the error rates. Checking each extracted command alone or after processing the whole utterance slightly improves both command recognition rates and also command recognition error rates, i.e., both by 1% absolute. Using plausibility values on semantic level and/or on word level from N-Best lists slightly decreases recognition rates, but decreases the error rates for both command and callsign extractions. If low error rates are more important than high recognition rates, plausibility values are another *must*.

Acknowledgment

The HAAWAII project and the solution PJ.10-W2-96 ASR have received funding from the SESAR Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 884287 and 874464, respectively.

References

[1] H. Helmke, M. Kleinert, N. Ahrenhold, H. Ehr, T. Mühlhausen, O. Ohneiser, L. Klamert, P. Motlicek, A. Prasad, J. Zuluaga Gomez, J. Dokic and E. Pinska Chauvin, "Automatic Speech Recognition and Understanding for Radar Label Maintenance Support Increases Safety and Reduces Air Traffic Controllers' Workload," in *15th USA/Europe Air Traffic Management Research and Development Seminar (ATM2023)*, Savannah, GA, USA, 2023.

[2] Deutsches Zentrum für Luft- und Raumfahrt (DLR), "HAAWAII: highly automated air traffic controller workstations with artificial intelligence integration," [Online]. Available: https://www.haawaii.de/wp/.

[3] B. Juang and L. Rabiner, "Automatic speech recognition -- a brief history of the technology development," in *Ga. Inst. Technol. Atlanta Rutgers*, Univ. Univ. California St. Barbara, 2005.

[4] D. Connolly, "Voice Data Entry in Air Traffic Control," in *Report N93-72621; National Aviation Facilities Experimental Center*, Atlantic City, NJ, USA, 1977.

[5] C. Hamel, D. Kotick and M. Layton, "Microcomputer System Integration for Air Control Training," in *Special Report SR89-01; Naval Training Systems Center*, Orlando, FL, USA, 1989.

[6] International Civil Aviation Organization (ICAO), "Doc 4444 ATM/501; ATM (Air Traffic Management): Procedures for Air Navigation Services," Montréal, QC, Canada, 2007.

[7] J. Cordero, N. Rodríguez, J. de Pablo and M. Dorado, "Automated Speech Recognition in Controller Communications applied to Workload Measurement," in *3rd SESAR Innovation Days*, Stockholm, Sweden, 26–28 November, 2013.

[8] S. Chen, H. Kopald, R. S. Chong, Y.-J. Wei and Z. Levonian, "Readback error detection using automatic speech recognition," in *12th USA/Europe Air Traffic Management Research and Development Seminar (ATM2017)*, Seattle, WA, USA, 2017.

[9] H. Helmke; K. Ondřej; S. Shetty; H. Arilíusson; T. S. Simiganoschi; M. Kleinert; O. Ohneiser; H. Ehr; J.-P. Zuluaga; P. Smrz, "Readback error detection by automatic speech recognition and understanding: results of HAAWAII project for Isavia's enroute airspace," in *12th SESAR Innovation Days*, Budapest, Hungary, 2022.

[10] V. Nguyen and H. Holone, "N-best list re-ranking using syntactic score: A solution for improving speech recognition accuracy in Air Traffic Control," in *16th International Conference on Control, Automation and Systems (ICCAS), Gyeong*, Gyeongju, Republic of Korea, 16–19 October, 2016.

[11] Y. Lin, "Spoken Instruction Understanding in Air Traffic Control: Challenge, Technique, and Application," *Aerospace 8(3),* March 2021.

[12] N. Ahrenhold, H. Helmke, T. Mühlhausen, O. Ohneiser, M. Kleinert, H. Ehr, L. Klamert and J. Zuluaga-Gómez, "Validating Automatic Speech Recognition and Understanding for Pre-Filling Radar Labels—Increasing Safety While Reducing Air Traffic Controllers' Workload," *Aerospace 10, 538.,* 2023.

[13] H. Helmke, H. Ehr, M. Kleinert, F. Faubel and D. Klakow, "Increased Acceptance of Controller Assistance by Automatic Speech Recognition," in *10th USA/Europe Air Traffic Management Research and Development Seminar (ATM2013)*, Chicago, IL, USA, 2013.

[14] H. Helmke, J. Rataj, T. Mühlhausen, Y. Oualil, M. Schulder, O. Ohneiser, H. Ehr and M. Kleinert, "Assistant-Based Speech Recognition for ATM Applications," in *11th USA/Europe Air Traffic Management Research and Development Seminar (ATM2015)*, Lisbon, Portugal, 2015.

[15] H. Helmke, O. Ohneiser, T. Mühlhausen and M. Wies, "Reducing controller workload with automatic speech recognition," in *IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, Sacramento, CA, US, 2016.

[16] H. Helmke, O. Ohneiser, J. Buxbaum and C. Kern, "Increasing ATM efficiency with assistant-based speech recognition," in *12th USA/Europe Air Traffic Management Research and Development Seminar (ATM2017)*, Seattle, WA, USA, 2017.

[17] M. Kleinert, H. Helmke, H. Ehr, C. Kern, D. Klakow, P. Motlicek, M. Singh and G. Siol, "Building Blocks of Assistant Based Speech Recognition for Air Traffic Management Applications," in *8th SESAR Innovation Days (SID 2018)*, Salzburg, Austria, 2018.

[18] M. Kleinert, H. Helmke, S. Moos, P. Hlousek, C. Windisch, O. Ohneiser, H. Ehr and A. Labreuil, "Reducing Controller Workload by Automatic Speech Recognition Assisted Radar Label Maintenance," in *9th SESAR Innovation Days (SID2019)*, Athens, Greece, 2019.

[19] H. Helmke, M. Slotty, M. Poiger, D. Ferrer Herrer, O. Ohneiser, N. Vink, A. Cerna, P. Hartikainen, B. Josefsson, D. Langr, R. García Lasheras, G. Marin, O.-G. Mevatne, S. Moos, M. N. Nilsson, M. Boyero Pérez, "Ontology for transcription of ATC speech commands of SESAR 2020 solution PJ.16-04," in *IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, London, United Kingdom, 2018.

[20] M. Kleinert, O. Ohneiser, H. Helmke, S. Shetty, H. Ehr, M. Maier, S. Schacht and H. Wiese, "Safety Aspects of Supporting Apron Controllers with Automatic Speech Recognition and Understanding Integrated into an Advanced Surface Movement Guidance and Control System," *Aerospace 2023, 10, 596,* 2023.

[21] H. Helmke, M. Kleinert, S. Shetty, O. Ohneiser, H. Ehr, H. Arilíusson, T. S. Simiganoschi, A. Prasad, P. Motlicek, K. Veselý, K. Ondřej, P. Smrz, "Readback error detection by automatic speech recognition to increase ATM safety," in *14th USA/Europe Air Traffic Management Research and Development Seminar (ATM2021)*, Virtual Conference, 2021.

[22] H. Helmke, O. Ohneiser, M. Kleinert, S. Chen, H. D. Kopald and R. M. Tarakan, "Transatlantic Approaches for Automatic Speech Understanding in Air Traffic Management," in *submitted to 15th USA/Europe Air Traffic Management Research and Development Seminar (ATM2023)*, Savannah, GA, USA, 2023.

[23] S. Chen, H. Helmke, R. Tarakan, O. Ohneiser, H. Kopald and M. Kleinert, "Effects of Language Ontology on Transatlantic Automatic Speech Understanding Research Collaboration in the Air Traffic Management Domain," *Aerospace 2023, 10, 526.,* 2023.

[24] J. Rataj, H. Helmke and O. Ohneiser, "AcListant with Continuous Learning: Speech Recognition in Air Traffic Control," *Air Traffic Management and Systems IV, Lecture Notes in Electrical Engineering,* pp. 93-109, 2021.

[25] CRIDA/ENAIRE, INDRA, DLR, NATMIG/SINTEF, LEONARDO and INTEGRA, "Deliverable D4.1.020 - PJ.10-W2-96 ASR-TRL6 Final TS/IRS - Part I," 21st March 2023.

[26] H. Kopald, "Automatic Speech Processing of United States ATC Speech - Large-scale processing of recorded real-world operations controller-pilot voice communications," in *Interspeech Satellite Workshop*, Brno, Czech Republic, 2021.

[27] M. Kleinert, S. Shetty, H. Helmke, O. Ohneiser, H. Wiese, M. Maier, S. Schacht, I. Nigmatulina, S. S. Sarfjoo and P. Motlicek, "Apron Controller Support by Integration of Automatic Speech Recognition with an Advanced Surface Movement Guidance and Control System," in *12th SESAR Innovation Days (SID 2022)*, Budapest, Hungary, 2022.

[28] O. Ohneiser, H. Helmke, S. Shetty, M. Kleinert, H. Ehr, Š. Murauskas, T. Pagirys, G. Balogh, A. Tønnesen, G. Kis-Pál, R. Tichy, V. Horváth, F. Kling, W. Rinaldi, S. Mansi and H. Usanovic, "Understanding Tower Controller Communication for Support in Air Traffic Control Displays," in *12th SESAR Innovation Days*, Budapest, Hungary, 2022.

[29] P. Motlicek, A. Prasad, I. Nigmatulina, H. Helmke, O. Ohneiser and M. Kleinert, "Speech Technologies to support Air Traffic Communication developed in HAAWAII," *accected at 13th SESAR Innovation Days*, Seville, Spain, Nov. 2023.