Reducing Controller Workload by Automatic Speech Recognition Assisted Radar Label Maintenance

Matthias Kleinert[1], Hartmut Helmke[1], Sylvain Moos[2], Petr Hlousek[3], Christian Windisch[4], Oliver Ohneiser[1], Heiko Ehr[1], Aline Labreuil[2]

[1]Institute of Flight Guidance, German Aerospace Center (DLR), Braunschweig, Germany,

[2]Thales LAS France SAS, Rungis, France, [3]Air Navigation Service Provider Prague, Czech Republic,

[4]Austro Control, Vienna, Austria

firstname.lastname@{dlr.de, thalesgroup.com, austrocontrol.at} and hlousek@ans.cz

***Abstract*—Various new hard- and software centered methods were recently implemented to replace paper flight strips through modern technical solutions. These solutions provide valuable information for other ATM applications about the verbally given guidance instructions from air traffic controllers (ATCos), but also tend to increase ATCo workload. Speech recognition applications, which automatically recognize and input the verbal ATCo instructions into a technical flight strip solution, can compensate the workload increase while maintaining the benefit for other ATM applications. Experiments from the AcListant® project in 2015 have shown that Assistant Based Speech Recognition (ABSR), which combines a conventional speech recognizer with an assistant system, can provide adequate recognition quality for the use in air traffic control (ATC) applications. AcListant® used a prototypic radar display implementation and a research prototype of a speech recognizer. This paper describes the exercise 220 of the SESAR 2020 funded solution PJ.16-04. It used a Commercial-Off-The-Shelf (COTS) speech recognition engine instead of a research prototype. Furthermore, a radar display developed by Thales Air Sys served for visualization. Command recognition rates varied greatly between 31% and 82% for different controllers. However, the concept from ABSR to predict possible ATCo instructions could be integrated with the COTS engine, which significantly decreased the command recognition error rate and led to a variation between only 4.8% and 6.6%, i.e. only a small amount of false recognitions were shown to the ATCo.**

Keywords—PJ.16-04, Assistant Based Speech Recognition, Automatic Speech Recognition, Checker, Air Traffic Control

I. INTRODUCTION

A. Problem

Recently, the Active Listening Assistant (AcListant®) project [1] has shown that a new type of Automatic Speech Recognition (ASR) [2] called Assistant Based Speech Recognition (ABSR) developed by Saarland University (UdS) and DLR [3]-[7] could be a solution to get acceptable recognition rates for ATCo to pilot voice commands. The Horizon 2020 funded project MALORCA (Machine Learning of Speech Recognition Models for Controller Assistance) successfully shows an approach to reduce the costs of ABSR environment adaptation [8]. It developed a basic ABSR architecture with generic building blocks and automatically adaptable models. For adaptation MALORCA greatly relies on Machine Learning techniques, therefore only few manual adaptations are necessary. Used airport frequencies, deviations from standard phraseology or specific acoustic and semantic variability, like accented speech of ATCos are automatically learned from recorded radar data and corresponding controller audio recordings, which are automatically transcribed (speech-to-text).

Although MALORCA's approach was successfully tested for two mid-size approach areas achieving command recognition error rates of 3.2% for Vienna and even 0.6% for Prague, there is still no implementation of the approach in an ops room. One reason is that MALORCA aimed to achieve Technology Readiness Level 1 (TRL-1), i.e. exploring the transition from scientific research to applied research by bringing together a wide range of stakeholders to investigate the essential characteristics and behaviors of applications, systems and architectures.

B. Solution

The experiments 220 of the SESAR 2020 funded solution PJ.16-04 CWP HMI (Controller Working Position Human Machine Interface) uses a commercial-off-the-shelf (COTS) speech recognizer engine, developed by Nuance Communications. The main focus of this ASR system is on dictation applications in various industry sectors. For the purpose of the PJ.16-04 project the grammar of this commercial speech recognizer was adapted to deliver command recognitions for ATCo instructions in the terminal maneuvering area (TMA) area.

Whatever the speech recognizer then delivered – based on the modified grammar – was validated against a set of predicted commands generated by a checker component from the MALORCA project before it was shown to the ATCo. The Human Machine Interface for the exercise was developed by Thales Air Sys, integrated into the SkyCentre in Rungis and validated with controllers from Austro Control and Air Navigation Services of the Czech Republic (ANS CR). The HMI is based on the Shape platform of Thales Air Sys and enables an easy integration into the operational TopSky system of Thales Air Sys after being validated by controllers. The HMI concept enables an easy correction of given ATC commands in the case that ASR fails to correctly recognize spoken ATC commands. In most cases, however, the recognition is correct, so that the ATCo saves cognitive resources for more important tasks than entering given commands into the aircraft radar label.

C. *Paper Structure*

In the next section we present related work with respect to machine learning and speech recognition applications in ATM. Section III describes the architecture used in exercise 220. Section IV explains the performed experiments, whereas section V presents the results. Section VI analyses the results and presents possible improvements before section VII summarizes and concludes.

II. Related Work

NASA reports spoken ATC word recognition already in 1977 [9]. Automatic Speech Recognition applications in the simulation environment for trainees to replace pseudo-pilots are used since the late 80s of the last century. ASR applications which should also understand air traffic controllers on the job had limited success.

One promising approach to improve ASR performance is using context knowledge regarding expected utterances. These attempts also go already back to the 80s [10], [11]. This information may heavily reduce the search space of the speech recognizer and lead to fewer missed recognitions [12]. Oualil et al. [13] analyzed the benefits of using context information for preprocessing versus using context for post-recognition. Helmke et al. extend the usage of context by generating the context from an assistance system, i.e. an Arrival Management System (AMAN), to support ABSR [4],[5]. ABSR can significantly reduce controllers' workload, which translates into fuel burn reduction and an increased runway throughput. These results were quantified for Dusseldorf approach area by performing baseline runs without and solutions runs with ABSR support: ATCo's clicking time reduced by a factor of three [6]. This workload reduction enables two landings more per hour and fuel burn reduction of approx. 60 liters per flight [7]. MALORCA project aims to automatically adapt the speech recognition building blocks to different approach areas. Automatic adaptation results for Vienna and Prague approach area from controller-pilot speech recordings and the corresponding radar tracks were presented in [14]. Command recognition error rates of the baseline system were reduced from 7.9% to below 0.6% for Prague and from 18.9% to 3.2% for Vienna using each time 18 hours of untranscribed speech recordings without silence.

III. Implementation of Functional Block "Automatic Speech Recognition" of EATMA Architecture

The MALORCA project developed an architecture to adapt ABSR to different approach areas, by defining generic building blocks and application dependent models, which are (as far as possible) trained by machine learning from recorded radar data and untranscribed ATCo speech recordings [15].

The ASR activity of solution PJ.16-04 created a functional block (FB) "Automatic Speech Recognition" into the European Air Traffic Management Architecture (EATMA) model, which is the common architecture framework for SESAR 2020. It is the sole mean of integrating the ATM operational and technical content developments produced by SESAR 2020 Projects in a consistent and coherent way [16]. The FB "Automatic Speech Recognition" mainly receives an audio signal as input and transforms it into a sequence of words, i.e. "speech-to-text". The sequence of words is than transformed further into a sequence of ATC concepts ("text-to-concepts") using a set of rules (an ontology) also defined by PJ.16-04 [17]. The resulting concepts can be used for further applications such as visualization on an HMI. The FB ASR consists according to EATMA of three ETAMA functions (Figure 1):

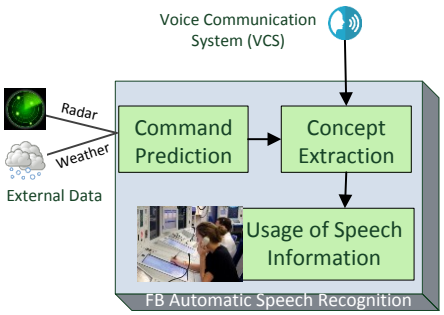

Figure 1. Integration of Automatic Speech Recognition into EATMA

1. **Command Prediction** to forecast possible future controller commands taking into account external data. This external data can be radar data, flight plan data, weather data, airspace data, and also historic data of those types.
2. **Concept Extraction**, which transforms the verbal controller utterance first into a sequence of words (speech-to-text) and then the word sequence into the corresponding ATC concepts, which are further combined to ATC commands (concepts-to-commands). The output of the Command Prediction can be used to ease both tasks and also for checking the extracted ATC commands against expectations (Checker task).
3. **Usage of Speech Information**: The extracted ATC commands are used to enable further applications at the same CWP HMI, e.g. callsign highlighting, command visualization, manual and automatic verification, workload estimation, pilot readback error detection.

The *Command Prediction* function was reused from the MALORCA project, where it is composed of the generic *Command Hypotheses Predictor* and an application dependent *Command Prediction Model (CPM)* [15]. MALORCA's overall architecture contains also the *Concept Extraction* function consisting of the

- **Generic building blocks:** Acoustic Recorder, Feature Extractor, ASR Decoder, N-Best-Generator, Corrector, Command Extractor, Command Filtering, Plausibility Checker and
- **Application dependent models:** ATC grammar, Domain Knowledge, Acoustic Model, Language Model and Lexicon.

In the described exercise the parts of the speech recognition related blocks and models of the *Concept Extraction* function

are based on a COTS Speech Recognition Engine from Nuance (Recognizer 10.2.4 with Nuance Speech Server 6, windows version on hardware with 16 GB RAM and SSD disk), so that most of the building blocks and models mentioned above are treated as black boxes, except the *Acoustic Model*, *ATC Grammar*, *Domain Knowledge* and *Plausibility Checker*. No special *Acoustic Model* was implemented to cover acoustic characteristics from the Czech respectively Austrian accent, i.e. a standard language pack from Nuance was used designed for speakers from the UK. The used *ATC grammar* is based on the ICAO phraseology plus some local deviations observed during the pre-trials so that a Czech version and an Austrian version in form of basic grammars exist. To continuously update the basic grammars during runtime the function *Command Prediction* generates a set of commands which are possible in the current situation according to the surveillance data, which consists of e.g. of radar data, flight plan and weather information. This than results in an update of the basic grammar and transforms it into its real time version like shown in Figure 2. The update cycle for command predictions and, therefore, also for grammar updates is 10 seconds.

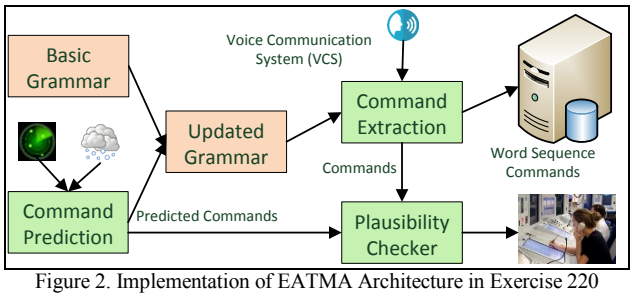

Figure 2. Implementation of EATMA Architecture in Exercise 220

The outputs of the Nuance Engines are a sequence of words and extracted ATC concepts given in an XML format. The concepts are transformed into a sequence of ATC commands by the *Command Extraction* function in Figure 2 according to the ontology [17]. The implementation used in validation trials takes into account the provided callsign information, QNH and frequency values to limit the possibilities of the speech recognition and *Command Extraction* output. Therefore, the output of the Nuance engine and the *Command Extraction* could never be a callsign or a QNH value which is not predicted. This limitation only represents a subset of the information provided by the *Command Prediction* function; the full set of predictions includes various command types with corresponding values, which are used by the *Plausibility Checker* in Figure 2. If a DESCEND command with flight level 70 is not predicted for a callsign, but it is the output of the *Command Extractor*, it will not be shown to the controller. If the rejected command was, however, said by the controller, we have a false rejection; otherwise we reduce the number of command recognition errors.

IV. Performed Exercises

This section describes the performed exercises with Prague and Vienna controllers in Thales SkyCentre Environment supported by DLR's Command Predictor function. First, the commonalities of both experiments are described. Section IV.A then describes the Prague approach and section IV.B the Vienna approach area exercise.

The reference scenarios always represent working without ASR support, i.e. the ATCo provides all the inputs into the system manually. In comparison the solution scenarios address the conditions when ASR is enabled. The system shall recognize which aircraft is contacted by the ATCo on the voice channel, highlight the appropriate callsign, translate the issued voice command(s) into system inputs and ensure that the highlighted callsign is retained, so that the ATCo can verify the recognized values. A special kind of command was the takeover of a flight from the previous managed sector, or its handover to the next sector for departures or to tower for arrivals. Such a command contained no value being shown to the ATCo in the radar label. Just the color coding of the callsign changed and the controller had time enough to reject the recognition before the recognition would result in a callsign transfer to the next ATCo position.

If the recognized and displayed values are correct, the ATCos didn't have to confirm the inputs. However, the ATCos were advised to correct all incorrect values manually. In addition to reference and solution runs one training run was performed per controller. The training run included parts where ASR was disabled (allowing the controllers to accustom to work on the simulation platform) and parts where ASR was enabled to familiarize controllers on how to interact with the ASR tool and its performance. Common for all exercises is:

- Inbounds and outbounds are considered, but overflights and also nearby airport traffic are ignored for simplicity.
- Typical traffic samples with typical Prague/Vienna callsign names are selected.

A. *Exercise 220a with Prague Controllers*

The validation scenarios mimicked the current traffic of Prague approach unit restricted to runway 24 (see Figure 3).

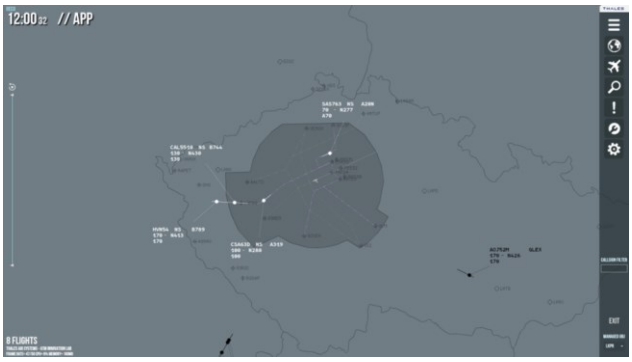

Figure 3. Approach Display in SkyCentre for Prague controller

The approach unit consists of seven working positions. The simulation, however, considers only the Departure Executive Controller (DEC), the Arrival Executive Controller (AEC) and the Director Executive Controller (PEC). In the case of lower traffic (up to 30 movement per hour), the positions AEC, PEC and DEC can be combined into a single position and all flights are controlled by one single Executive Controller. This option

ISSN 0770-1268

was chosen for the validation exercises, enabling more simulations with the same budget. The situation also often appeared when data for command prediction training was recorded during the MALORCA project. So the command prediction model (CPM) from the MALORCA project could be reused [14]. Two scenarios were created, (1) a medium traffic scenario containing 24 movements per hour and (2) a heavy traffic scenario with 30 movements per hour.

B. *Exercise 220b with Vienna Controllers*

The validation scenarios for Vienna modelled the current traffic of Vienna approach unit restricted to only runway 34 (see Figure 5). There are in total 10 working positions within Vienna approach. The simulation, however, considers only the positions NERDU-Sector Executive, MABOD-Sector Executive, BALAD-Sector Executive, and Feeder for runway 34 Executive Controller (also called Director). The sector positions are shown in Figure 4.

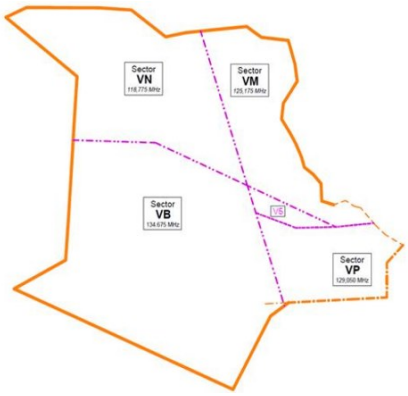

Figure 4. BALAD (VB), NERDU (VN), MABOD (VM) sectors of Vienna

Each day of the final validation trials only one Austro Control ATCo was available. Therefore, the duties of all four positions were performed by one ATCo. Traffic flows were adapted accordingly so that traffic was manageable. This situation, however, never happened when data for command prediction training was recorded in 2016 for the MALORCA project. Therefore, the CPM from MALORCA could not be used. A complete retraining of the CPM was necessary. Training data for this was recorded during pre-trials in March 2019 resulting in 3,400 different commands.

Figure 5. Approach Display in SkyCentre for Vienna controller

C. *Validation hypotheses*

The validation objectives are to demonstrate ASR technical feasibility and interoperability (TFI), ASR performance stability (PST), ASR operational feasibility (OPF), Human performance (HUP), Safety (SAF), and TMA capacity (CAP). Figure 6 shows the dependency graph of these validation objectives.

Different validation success criteria were formulated for each of the six validation objectives resulting in validation hypotheses. For *ASR performance stability (PST)* the following hypotheses were formulated:

- The ASR performance is maintained during the time, i.e. Command Recognition Rate (CmdRR) greater than 85% and Command Recognition Error Rate (CmdER) less than 2.5%.
- The ASR command recognition rate is acceptable and there is no major difference between clearances types.

More validation hypotheses are presented in the next section together with the results.

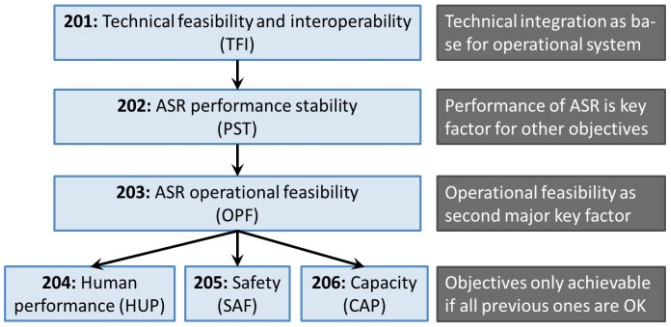

Figure 6. Dependancy Graph of Validation Objectives

V. Validation Results

The validation runs with four controllers from Prague took place in Rungis in February 2019, the runs with two Vienna controllers in May 2019. Each controller participated in four runs (medium/heavy with and without ASR support) plus the training runs.

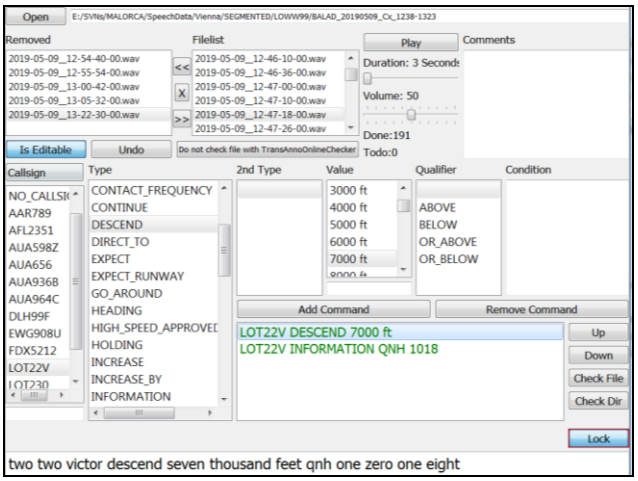

Figure 7. Screen Dump of Transcription and Annotation Tool CoCoLoToCoCo

ISSN 0770-1268

The audio files of all runs were recorded together with the corresponding radar data. Additionally the predicted commands were saved when an audio file was recorded. In total we got 16 Prague and 8 Vienna runs (without training runs). All 6,600 commands from the runs were manually transcribed and annotated by DLR with CoCoLoToCoCo tool (Controller Command Logging Tool for Context Comparison), which already implements the ontology developed by PJ.16-04 [17], see Figure 7. A run of 45 minutes takes approximately 90 – 120 minutes of transcription and annotation time including the time for error corrections.

A. *Objective Prague Results*

The following Table 1 provides command recognition rates etc. for all eight solution runs. Row "All commands" contains the results when considering all 1,994 commands annotated. Row "> 10 commands" excludes command types which occurred less than 10 times (e.g. type NAVIGATION_OWN or REPORT). Row "Must" contains only types, which were defined in Functional Requirements Document (FRD) as "recognition must be supported" (e.g. all command types which are also shown in the ATCo's HMI). Row "Must+Should" shows the results, which should or must be recognized according to FRD (handover type is e.g. a "should", but not a "must" type).

TABLE 1: ASR PERFORMANCE OF ALL SOLUTION RUNS FOR PRAGUE

	totalGiven	RecognRate	ErrorRate	RejectRate
All commands	1994	70.41%	15.78%	14.75%
Must	1461	72.67%	17.23%	12.64%
Must+Should	1646	72.17%	15.72%	14.37%
> 10 commands	1989	70.69%	14.76%	15.34%

Column "totalGiven" counts the number of commands, given/annotated in all the eight solution runs. Column "RecognRate" contains the average command recognition rate: We calculate the average command recognition rate for each of the eight solution runs and then the average, so that each solution run has the same weight for the average value. A command is correctly recognized if the callsign AND command type AND value AND qualifier are correctly recognized. The correct or wrong recognition of the unit has no effect on the result. Column "ErrorRate" provides the average command recognition error for the eight runs. Column "RejectRate" provides the average command rejection rates. A command is counted as rejected if nothing is recognized or the output was NO_CALLSIGN or NO_CONCEPT (NO_CALLSIGN or NO_CONCEPT are not correct according to annotation). Details of these calculations are provided in [4].

TABLE 2: ASR PERFORMANCE FOR PRAGUE WHEN CHECKER IS USED

	With Checker			Without Checker	
	totalGiven	RecognRate	ErrorRate	RecognRate	ErrorRate
All commands	1994	70.36%	5.36%	70.41%	15.78%
Must	1461	72.61%	5.32%	72.67%	17.23%
Must+Should	1646	72.11%	4.84%	72.17%	15.72%
> 10 commands	1989	70.65%	5.32%	70.69%	14.76%

The results in Table 1 are still without using the Plausibility Checker (see Figure 2). Table 2 shows the results when the Plausibility Checker is used, i.e. commands which are not predicted are than rejected as well. The last two rows repeat the results of Table 1, i.e. when the Checker is not used.

The Checker sometimes also rejects correct recognitions. In these cases the controller uses commands which are not predicted. The resulting reductions in the recognition rates, however, are neglectable compared to the dramatic reduction (factor of 3) of the command recognition error rate.

In addition to the manual quite work intensive post-run annotation, a first rough analysis was manually performed by a subject matter expert during the validation runs. The following Table 3 shows the results of this analysis per ATCo.

TABLE 3: CORRECTLY DISPLAYED COMMANDS ON RADAR SCREEN OF ATCO

	Medium Scenario	Heavy Scenario	Both Scenarios
ATCO 1	76.3%	70.8%	73.7%
ATCO 2	78.5%	84.3%	81.7%
ATCO 3	80.1%	76.5%	78.3%
ATCO 4	78.1%	79.7%	79.0%

The recognition rates online calculated via drawing stroke on a piece of paper during the solution runs, seems to be much higher than the offline calculated rates after the runs. However, the subject matter expert only considers the relevant commands, which are shown on the radar label of the ATCo. Heading commands are a second reason. The controller only sees the heading value, but not the qualifier (LEFT/RIGHT), which was often wrongly recognized by the Nuance engine. The subject matter expert also has to decide at once and cannot replay the utterance.

TABLE 4: COMMAND PREDICTION ACCURACY FOR PRAGUE APPROACH

	With Checker				
	totalGiven	RecognRate	ErrorRate	# CmdE	CmdPER
All commands	1994	70.36%	5.36%	6	0.31%
Must	1461	72.61%	5.32%	2	0.15%
Must+Should	1646	72.11%	4.84%	2	0.13%
> 10 commands	1989	70.65%	5.32%	2	0.11%

Table 4 provides in the last two columns the accuracy of the Command Hypotheses Predictor and compares them to the command recognition rate and command recognition error rate, e.g. only two "Must" commands out of 1,461 commands given by the ATCo were not predicted (0.15%).

B. *Objective Vienna Results*

The following Table 5 provides the recognition results of all four solution runs for the Vienna ATCos, when the Plausibility Checker is not used.

TABLE 5: ASR PERFORMANCE OF ALL SOLUTION RUNS FOR VIENNA

	totalGiven	RecognRate	ErrorRate	RejectRate
All commands	1259	54.77%	20.39%	25.69%
Must	1094	54.42%	22.07%	25.54%
Must+Should	1094	54.34%	22.16%	25.53%
> 10 commands	1251	55.45%	16.91%	28.33%

Although the total number of commands is equal for "Must" and "Must+Should" type the resulting rates are different: Type "INFORMATION ATIS" is e.g. a should-type for Vienna. If recognized (but not said), it is thrown out from the recognition and calculated neither as an error nor as a rejection. Table 6 shows the results when Plausibility Checker supports

ISSN 0770-1268

the Vienna controllers and compares the results again, in the last columns, to the situation without checker. The high reduction in the command recognition rate again enables a very significant improvement of the command recognition error rate.

TABLE 6: ASR PERFORMANCE FOR VIENNA WHEN CHECKER IS USED

		With Checker		Without Checker	
	totalGiven	RecognRate	ErrorRate	RecognRate	ErrorRate
All commands	1259	52.28%	6.14%	54.77%	20.39%
Must	1094	53.29%	6.56%	54.42%	22.07%
Must+Should	1094	53.22%	6.65%	54.34%	22.16%
> 10 commands	1251	52.93%	5.84%	55.45%	16.91%

Table 7 shows the online annotation of the subject matter expert. It shows again the already explained differences between offline evaluation and online evaluation taking the ATCo's feedback on the HMI into account, but it also shows the big difference between the two ATCos from Vienna, whereas the differences between the four different Prague ATCos are much smaller (Table 3). Vienna controller deviated much more in used ICAO phraseology.

TABLE 7: CORRECTLY DISPLAYED COMMANDS ON RADAR SCREEN OF ATCO

	Medium Scenario	Heavy Scenario	Both Scenarios
ATCO 1	78.2%	78.3%	78.2%
ATCO 2	44.5%	60.5%	52.5%

Table 8 shows the accuracy of the Command Hypotheses Predictor for Vienna Approach in the last two columns.

TABLE 8: COMMAND PREDICTION ACCURACY FOR VIENNA APPROACH

	With Checker				
	totalGiven	RecognRate	ErrorRate	ctxErrCnt	ctxErrRate
All commands	1259	52.28%	6.14%	58	4.77%
Must	1094	53.29%	6.56%	27	2.56%
Must+Should	1094	53.22%	6.65%	27	2.56%
> 10 commands	1251	52.93%	5.84%	53	4.33%

Compared to Prague data (Table 4) command prediction accuracy seems to be improvable for Vienna. However, Prague CPM model could be reused from MALORCA. Vienna CPM model was retrained on data obtained in pre-trails, because the combined airspace does not exist in real-life data.

C. *Validating the Validation Hypotheses*

Although the validations were performed in a laboratory environment, the previous sections show that both the command recognition rates and also the command recognition error rates are improvable and are below the results of the MALORCA project. MALORCA also uses Prague and Vienna approach as validation airports. MALORCA, however, trained the acoustic models for Czech and Austrian accent. MALORCA achieved a command recognition rate of 91.7% for Prague and 85.2% for Vienna. The command recognition error rates were 0.6% for Prague and 3.2% for Vienna. MALORCA's baseline system – with 18 hours of untranscribed data of each airport also achieves command recognition rates of only 79% for Prague and 60% for Vienna. The numbers presented in the previous sections together with subjective ATCo's feedback obtained via questionnaires was used to validate the six validation objectives presented already in Figure 6.

TABLE 9: VALIDATION OBJECTIVES AND HYPOTHESES FOR TFI, PST

Obj	Validation Hypothesis	Prague	Vienna
TFI	The ATCos are able to use the ASR system without visible slowing down of the system or malfunctioning compared to basline	OK	OK
PST	CmdRR > 85% and CmdER < 2.5%, CpER < 10%	POK	POK
	The ASR command recognition rate is acceptable and there is no major difference between clearances types.	NOK	NOK

The results in Table 9 show that it is feasible to integrate ASR into a controller working position without influencing its performance (objective TFI). The abbreviation in the last two columns of this and the following tables stand for (1) OK = Achieved, (2) POK = Partially Achieved and (3) NOK = Not Achieved. As explained in the beginning of the section the performance stability objective (PST) was not achieved. The improvable recognition performance is also reflected in the ASR operational feasibility (OPF) validation objective (Table 10).

TABLE 10: VALIDATION OBJECTIVES AND HYPOTHESES FOR OPF

Obj	Validation Hypothesis	Prague	Vienna
OPF	The ASR will support the performance of operations.	NOK	POK
	The ASR will be adequate for the accomplishment of operations with respect to CmdRR and CmdER	NOK	NOK
	ASR supports the performance of operations in terms of timeliness	NOK	NOK
	The number of error is within tolerable limits (CmdER < 2.5%)	NOK	NOK

TABLE 11: VALIDATION OBJECTIVES AND HYPOTHESES FOR HUP, PART 1

Obj	Validation Hypothesis	Prague	Vienna
HUP	The level of Command Recognition Rate will be >= 85%.	NOK	NOK
	The responsiveness is adequate (< 2 seconds)	NOK	POK
	Tasks can be achieved in timely manner.	NOK	NOK
	The level of workload with the introduction of ASR is maintained at the acceptable level.	OK	OK
	Situation Awareness (SA) is not reduced.	OK	OK
	The number of severe human errors is within tolerable limits.	OK	NOK

TFI, PST and OPF validation objectives are a pre-condition for the other three validation objectives. Therefore, they are also not fully achieved. Human performance is shown in Table 11 and Table 12. Although recognition performance is not acceptable, controllers' feedback in questionnaires was not completely negative. Mostly they see that radar label maintenance support via ABSR could reduce their workload, if recognition performance is significantly improved. Due to bad recognition performance, also no capacity improvements were observed (Table 13).

ISSN 0770-1268

TABLE 12: VALIDATION OBJECTIVES AND HYPOTHESES FOR HUP, PART II

Obj	Validation Hypothesis	Prague	Vienna
HUP	The design of the user interface supports ATCos in carrying out the tasks.	OK	OK
	The design of the user interface supports ATCos in carrying out the tasks.	OK	OK
	The presentation of information supports ATCos in detection of possible system errors.	OK	OK
	The level of trust in the ASR system is acceptable.	OK	OK
	The ASR is assessed as acceptable.	NOK	NOK
	The level of Command Recognition Rate will be >= 85%.	NOK	NOK
	The responsiveness is adequate (< 2 seconds)	NOK	POK
	Tasks can be achieved in timely manner.	NOK	NOK

TABLE 13: VALIDATION OBJECTIVES AND HYPOTHESES FOR CPA AND SAF

Obj	Validation Hypothesis	Prague	Vienna
CAP	The workload reduction provided by ASR system is adequate to increase ATM capacity.	NOK	NOK
SAF	The completeness and accuracy of the information provided by the ASR is adequate.	NOK	NOK
	The responsiveness of the ASR is adequate	NOK	NOK
	The number or severity of errors resulting from the introduction of ASR is within tolerable limits.	NOK	NOK
	The recovery means for errors resulting from ASR are identified to minimize operational impact	OK	OK

VI. INTERPRETATION OF RESULTS AND NEXT STEPS

ATCos' feedback also addresses the recognition times (Table 11). The delays between issuing a clearance and feedback on the HMI were sometimes too long. ATCos many times thought that the commands were not recognized and start to manually input the commands whereas in parallel the values are displayed in the HMI. This is due to the implementation. Recognition starts first, when the ATCo has released the push-to-talk button. This, however, is more a software design than an ABSR issue. Recognition could already start when ATCo starts talking and intermediate results could be provided if the controller gives multiple commands to the pilot. This is already implemented in the Nuance recognition engine 11.

Correcting wrongly recognized commands takes too much of ATCos capacity even with only medium traffic. The following order describes all cases identified from less intrusive to very intrusive in the ATCos work:

- No recognition: ATCo needs to enter complete command into radar label, i.e. same situation as without ABSR.
- Wrong callsign recognized and no command displayed: Same as for "no recognition". No correction for wrong callsign necessary, but callsign highlighting of wrong aircraft demands cognitive ATCo resources.
- Wrong value for correct callsign with correct command type: e.g. DESCEND 210 FL instead of DESCEND 200 FL: ATCo needs to correct the wrong value in the radar label. Requires cognitive resources to identify wrong value. Risk is that a wrong value is not recognized by ATCo.
- Wrong command type for correct callsign recognized: e.g. HEADING 200 LEFT instead of REDUCE 200 kt: ATCo needs to delete the wrong recognition and also enter the correct recognition. Additional risk is that misrecognition is not detected.
- Command wrongly/correctly recognized for wrong callsign: ATCo needs to delete recognition for wrong callsign, identify the position of the correct callsign and enter the correct commands in the radar label. Additional risk is that misrecognition for wrong callsign is not detected.

This results in the summary that *no recognition is better than a wrong recognition*. There is of course always a trade-off between recognition (CmdRR) and error rate (CmdER).

Even though Prague controllers use a reduced phraseology subset recognition rates for them are also quite low. Table 14 shows that Prague controllers use much smaller range of different words than Vienna ones. Therefore, the modelled phraseology needs to be improved.

TABLE 14: COMMAND COMPLEXITY

		Number of Different Words
Prague	ATCo1	120
	ATCo2	125
	ATCo3	124
	ATCo4	118
Vienna	ATCo1	146
	ATCo2	167

Trainees can be forced to strictly follow ICAO phraseology. However, ATCos already on the job will never accept a system which does not support their current phraseology, although AcListant®-Strips project has shown that controllers will more and more adapt their phraseology towards modelled phraseology if they get benefits, i.e. better ASR support. Nevertheless, we first need to improve ASR performance, and then ATCos might slightly adapt their phraseology and not the other way round!

The drawback, however, is that then an acoustic model and a grammar are available for the lab environment, but at the end the benefits are in real life traffic. Therefore, no time should be wasted for improving the models on laboratory data, but on real life data. Thousands of hours of training data from the ops room for ABSR model training are available nearly for free, provided that data privacy issues are solved. For lab data costly experiments are necessary just for generating training data.

On the other hand special situations like near misses and thunderstorm weather and heavy traffic scenarios could be created in the lab environment. These situations are difficult to

produce in real life and are not desired, i.e. validations with special situation scenarios need to be done (in the lab environment).

VII. CONCLUSIONS

The experiment 220 of SESAR 2020 funded solution PJ.16-04 validated a radar display developed by Thales Air Sys and a Commercial-Off-The-Shelf (COTS) speech recognition engine. Command recognition rates varied between 31% and 82% for different controllers. The ABSR concept to use a Plausibility Checker based on predicted possible controller commands could dramatically reduce the command recognition error rate ranging between [14.7% .. 22.6%] to a range of [4.8% .. 6.6], i.e. most of the false recognitions were not shown to the ATCo.

Based on the work performed and the results of the validation exercise, the following recommendations are issued in view of further research and implementation work:

- Reduce the delay until the voice communication and its recognition is displayed in the HMI by processing the recognition already during the communication.
- Use special acoustic model trained for final end users, i.e. English with Czech accent for Czech and English with Austrian accent for Austrians etc.
- Extend the grammar to support more phraseology deviations.
- Extend ABSR systems also to other ATC workstations such as in tower and remote tower environment [18].
- Training of acoustic and language model (grammar) should be done with real-life data and on automatically transcribed voice recordings (MALORCA approach).
- Larger amount of runs should be performed in order to achieve a higher level of significance of the results, provided that the identified improvements result in better ASR command recognition rates, which are comparable to AcListant®-Strips [7] and MALORCA [14] project.

ACKNOWLEDGMENT

We would like to thank all the controllers and pseudo-pilots who participated in the pre-trials and in the final validations trials.

REFERENCES

[1] The project AcListant® (Active Listening Assistant) http://www.aclistant.de/wp, n.d.

[2] T. Shore, F. Faubel, H. Helmke, and D. Klakow, "Knowledge-based word lattice rescoring in a dynamic context," Interspeech 2012, Sep. 2012, Portland, Oregon.

[3] 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.

[4] H. Helmke, J. Rataj, T. Mühlhausen, O. Ohneiser, H. Ehr, M. Kleinert, Y. Oualil, and M. Schulder, "Assistant-based speech recognition for ATM applications," in 11th USA/Europe Air Traffic Management Research and Development Seminar (ATM2015), Lisbon, Portugal, 2015.

[5] O. Ohneiser, H. Helmke, H. Ehr, H. Gürlük, M. Hössl, T. Mühlhausen, Y. Oualil, M. Schulder, A. Schmidt, A. Khan, and D. Klakow, "Air Traffic Controller Support by Speech Recognition," in N. Stanton, S. Landry, G. Di Bucchianico, and A. Vallicelli (Eds.), Proceedings of the 5th International Conference on Applied Human Factors and Ergonomics AHFE 2014, Advances in Human Aspects of Transportation: Part II, pp. 492-503, Krakow, CRC Press, 2014.

[6] H. Helmke, O. Ohneiser, Th. Mühlhausen, M. Wies, "Reducing controller workload with automatic speech recognition," in IEEE/AIAA 35th Digital Avionics Systems Conference (DASC). Sacramento, California, 2016.

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

[8] The project MALORCA (Machine Learning of Speech Recognition Models for Controller Assistance) http://www.malorca-project.de, n.d.

[9] D. W. Connolly, "Voice data entry in air traffic control," NADC, Proceedings: Voice Technology for Interactive Real-Time Command Control Systems Application, 1977, pp 171-196

[10] S.R. Young, W.H. Ward, and A.G. Hauptmann, "Layering predictions: Flexible use of dialog expectation in speech recognition," in Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI89), Morgan Kaufmann, 1989, pp. 1543-1549.

[11] S.R. Young, A.G. Hauptmann, W.H. Ward, E.T. Smith, and P. Werner, "High level knowledge sources in usable speech recognition systems," in Commun. ACM, vol. 32, no. 2, Feb. 1989, pp. 183-194.

[12] D. Schäfer, "Context-sensitive speech recognition in the air traffic control simulation," Eurocontrol EEC Note No. 02/2001 and PhD Thesis of the University of Armed Forces, Munich, 2001.

[13] Y. Oualil, M. Schulder, H. Helmke, A. Schmidt, and D. Klakow, "Real-Time Integration of Dynamic Context Information for Improving Automatic Speech Recognition," Interspeech, Dresden, Germany, 2015.

[14] M. Kleinert, H. Helmke, G. Siol, H. Ehr, A. Cerna, C. Kern, D. Klakow, P. Motlicek et al., "Semi-supervised Adaptation of Assistant Based Speech Recognition Models for different Approach Areas," in IEEE/AIAA 37th Digital Avionics Systems Conference (DASC). London, England, 2018.

[15] M. Kleinert, H. Helmke, H. Ehr, Chr. Kern, D. Klakow, P. Motlicek, M. Singh, and G. Siol, "Building Blocks of Assistant Based Speech Recognition for Air Traffic Management Applications," 8th SESAR Innovation Days, Salzburg, 2018.

[16] SESAR, "European ATM Architecture (EATMA), One Framework, One Plan," https://www.sesarju.eu/sites/default/files/documents/ projects /EATMA_WAC-Factsheet-online.pdf.

[17] H. Helmke, M. Slotty, M. Poiger, D. F. Herrer, O. Ohneiser et al., "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.

[18] O. Ohneiser, H. Helmke, M. Kleinert; G. Siol, H. Ehr, S. Hobein, A.-V. Predescu, J. Bauer, "Tower Controller Command Prediction for Future Speech Recognition Applications," 9th SESAR Innovation Days, Athens, Greece, 2019.