Measuring Speech Recognition And Understanding Performance in Air Traffic Control Domain Beyond Word Error Rates

Hartmut Helmke, Shruthi Shetty, Matthias Kleinert,
Oliver Ohneiser, Heiko Ehr
Institute of Flight Guidance,
German Aerospace Center (DLR),
Braunschweig, Germany,
FirstName.LastName@dlr.de

Amrutha Prasad, Petr Motlicek
Idiap Research Institute,
Martigny, Switzerland,
amrutha.prasad@idiap.ch, petr.motlicek@idiap.ch

Aneta Cerna
Air Navigation Service Provider Czech Republic,
Jenec, Czech Republic
cernaa@ans.cz

Christian Windisch
Austro Control, Wien, Austria,
Christian.Windisch@austrocontrol.at

***Abstract*—Applying Automatic Speech Recognition (ASR) in the domain of analogue voice communication between air traffic controllers (ATCo) and pilots has more end user requirements than just transforming spoken words into text. It is useless for, e.g., readback error detection support, if word recognition is perfect, as long as the semantic interpretation is wrong. For an ATCo it is of almost no importance if the words of a greeting are correctly recognized. A wrong recognition of a greeting should, however, not disturb the correct recognition of, e.g., a "descend" command. More important is the correct semantic interpretation. What, however, is the correct semantic interpretation especially when ATCos or pilot, deviate more of less from published standard phraseology? For comparing performance of different speech recognition applications, 14 European partners from Air Traffic Management (ATM) domain have recently agreed on a common set of rules, i.e., an ontology on how to annotate the speech utterances of an ATCo on semantic level. This paper first presents the new metric of "*unclassified word rate*", extends the ontology to pilot utterances, and introduces the metrics of command recognition rate, command recognition error rate, and command recognition rejection rate. This enables the comparison of different speech recognition and understanding instances on semantic level. The implementation used in this paper achieves a command recognition rate better than 96% for Prague Approach, even if word error rate is above 2.5% based on more than 12,000 ATCo commands – recorded in both operational and lab environment. This outperforms previous published rates by 2% absolute.**

Keywords: word error rate, command recognition rate, language understanding, air traffic control, ATC, unclassified word rate

I. INTRODUCTION

Nowadays, enhanced Automatic Speech Recognition (ASR) systems are used in Air Traffic Control (ATC) training simulators to replace expensive simulation pilots. This work has started already in the late 80s [1]. Although ASR systems are widely used in everyday life (e.g., Siri®, Alexa®) and ATC phraseology is standardized [2], recognizing and understanding controller-pilot communication is still a big challenge and not solved with satisfactory performance. Due to lack of ATC-specific training data, current ASR systems still face challenges with specialized ATC vocabulary and syntax, controllers' deviations from the standard phraseology, and a variety of speakers and accents [3]. Cordero et al. (2012) reported WER (= word error rate) of more than 80% with different Commercial-off-the-shelf (COTS) recognizers [4].

Different metrics exist to evaluate the performance of ASR. The most widely used metric in ASR applications is the WER based on the Levenshtein distance [5]. However, the decision makers of air navigation service providers (ANSPs) are not primarily interested in these low-level metrics. They are interested in reducing costs and efforts. The AcListant®-Strips project quantified the benefits of using speech recognition with respect to both efficiency and ATCo workload: The ATCo workload for radar label maintenance could be reduced by a factor of three [6] and the support of ASR enabled fuel savings of 50 to 65 liters per flight [7].

In this paper we will concentrate on the semantic level, i.e. on **annotations**, to evaluate the ASR and speech understanding performance in ATC domain. The concentration on the semantic is best illustrated by an example with two **transcriptions** for ATCo utterances:

- "good morning lufthansa two bravo alfa radar contact descend flight level eight zero and speed two two zero knots",
- "bravo alfa identified two twenty knots descend level eighty".

On word level there is a large difference between the two transcriptions, but semantically they have the same meaning. According to the ontology defined by various European partners from the ATM industry and research [8], both transcriptions correspond to the following three ATC commands: "DLH2BA INIT_RESPONSE, DLH2BA DESCEND 80 FL, DLH2BA SPEED 220 kt", but provided in a different order.

The ontology rules enable the comparison of different speech recognition and understanding systems for ATC application on a semantic level by considering each ATC command (e.g., DLH2BA SPEED 220 kt) as one (big) entity, i.e., word, and calculating the Levenshtein distances w.r.t. the gold annotations. Gold transcriptions or annotation, respectively refer to the manually checked transcriptions/ annotations, i.e. the ones, which are assumed to be correct.

The following section gives a brief overview of related work with respect to transcription and annotation in the ATC domain. Section 3 introduces the main elements of the ontology for ATC command annotation and describes the enhancements of the ontology with respect to annotated pilot utterances. Section 4 presents the suggested metrics for evaluation of speech recognition and understanding systems in ATC. Section 5 presents evaluation results from different projects, followed by a conclusion.

II. Related Work

One of the first publicly available corpora with transcribed speech recordings was the LDC94S14A data set. The audio files are 8 kHz, 16-bit linear sampled data, representing continuous monitoring, without squelch or silence elimination, of a single FAA frequency for one to two hours [9]. A European data set for the ATC domain is the Air Traffic Control Simulation (ATCOSIM) Speech corpus. It is a speech database containing ATCo utterances created during ATC real-time simulations at EUROCONTROL in Brétigny [10]. Our transcription rules for writing down ATC utterances word by word are very similar, but in addition to [10] we also propose rules for annotation. Nguyen and Holone [11], [12] proposed 10 classes to replace word sequences with their corresponding class label, e.g., callsign, unit-name, fix, number. Johnson et al. [13] propose a keyword and value representation in JSON format [14], where keywords could be Callsign, ToFix, FlightLevel, Altimeter, etc.

In the AcListant® project [15], Saarland University and DLR created an ontology which consists of four elements, i.e., callsign, command type, command value, and unit [16], [17]. This approach already covering more than 30 commands reached its limits in the *MALORCA* project [18]. Here voice recordings from Prague and Vienna live traffic were integrated. An increasing number of command types (e.g., QNH, INFORMATION, REPORT_SPEED, EXPECT_RUNWAY) had to be supported. Additionally, conditional clearances were modelled [19]. In 2002, NATS analyzed possible applications of ASR within the London Terminal environment [20]. Initially, several ontologies were proposed based on a statistical Language Model (LM). At project closure, the ontology encompassed five elements: callsign, standard type, non-standard type, value, and type unit (e.g., feet, degrees).

The SESAR funded solution PJ.16-04 of the project Controller Working Position Human Machine Interface (CWP HMI) tried to harmonize all these approaches. 22 partners from European ATM industry, research, and from air navigations providers agreed on a so-called ontology, i.e., a set of rules, for command annotations [8]. It is not final yet, which means that updates/changes are still expected. The projects STARFiSH [21] and "HMI Interaction Modes for Airport Tower" [22] expand the ontology with respect to ATC ground and tower commands including remote tower operations, based on the work of Ohneiser et al. [23] with results for Lithuanian and Hungarian remote tower environment [24]. An ontology for tower commands was also used by Chen et al. [25], when ASR was used to automatically detect read back errors of ground traffic.

The projects "HMI Interaction modes for approach control" [26] and the SESAR funded project HAAWAII [27] also include pilot utterances as well as enroute and oceanic traffic. Some of these extensions are presented in the next section. Further ASR projects require an even bigger variety of annotated commands, i.e., greetings become important for workload estimation even if they have hardly any concrete meaning for ATC communication content. Greetings are normally uttered in low workload situations and observed less frequently in high workload situations, a hypothesis further analyzed by the HAAWAII project. The first example above is then transformed into "DLH2BA GREETING, DLH2BA INIT_RESPONSE, …".

III. Ontology for Annotation of ATC Utterances

A subset of the CWP HMI ontology [8] for annotation with new elements *speaker* and *reason* is presented in the next subsection III.A. The ontology is being extended in the SESAR funded HAAWAII project as shown with detailed examples in subsection III.B.

A. Basic Annotation Ontology Structure

The rules define that an utterance consists of one or more instructions (Figure 1) and each instruction starts with the callsign, even if the callsign is only said once. The full intended callsign (from the flight plan or surveillance data) is provided, i.e., AUA123B is used even if only "austrian three bravo" is said or recognized. This compensates for misrecognitions on word level and also deals with commonly used abbreviations for callsigns in ATC. If no callsign is said or could not be uniquely determined, "NO_CALLSIGN" is used. Figure 1 depicts the structure of an instruction and shows that an instruction consists of a callsign, a command, and optional conditions.

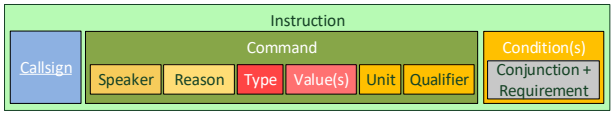

Figure 1: Elements of an instruction consisting of a callsign, a command, and condition(s).

A command always has a type, which determines the number of allowed values. The command type is composed of two elements: command first type and an optional second type. Additional optional fields are unit (e.g., FL, ft, kt), qualifier (e.g., LESS, OR_BELOW, LEFT), speaker (PILOT or empty), and reason (REQUEST, REPORTING or empty for, e.g., readbacks and commands).

Various examples from different application areas should illustrate the agreed rules. For approach traffic *"speed bird six nine six victor keep speed one eight zero knots until five miles final"* would result in "BAW696V MAINTAIN SPEED 180 kt UNTIL 5 NM FINAL". The last four elements after unit "kt"

are the conditional clearance with the conjunction "UNTIL". Here, we see that the type consists of two words (e.g., MAINTAIN SPEED) where MAINTAIN is the first command type and SPEED is the second type. However, command types such as CLIMB, DESCEND, SPEED, etc. consist of only the first type. An example from the tower is *"lufthansa four nine eight taxi to alfa four eight via lima and november seven"*, which corresponds to "DLH498 TAXI TO STAND_A48" and "DLH499 TAXI VIA TX-L TX-N7". Here, the command type "TAXI TO" can only have one value, whereas multiple values are allowed for "TAXI VIA". The ontology requires a configuration file, which defines that the word sequence *"alfa four eight"* is mapped to "STAND_A48" and that *"lima"* in a TAXI VIA command is mapped to "TX-L". This eases the maintenance of the semantic extractor, if new waypoints, airline designators or taxiways need to be added and also enable easy adaptation to other airports even by ATC experts who are not directly involved in ASR development.

B. *Enhancement Examples of Annotation Ontology*

In the following transcription and annotation examples we use colors to support the reader in immediately recognizing the different ATC concepts in an utterance transcription and in the corresponding annotation. The colors are independent from those used in Figure 1. It also shows the simplicity of the ontology rules.

The following utterance considers both pilot and ATCo utterances for enroute traffic: *"Pilot: reykjavik control [NE Icelandic] godan dag [/NE] ice air six eight kilo passing level one eight zero climbing two nine zero ATCo: [unk] six eight kilo reykjavik control [NE Icelandic] godan dag [/NE] identified climb to flight level three six zero"*. Here, "NE" represents "Non-English" text. The transcription rules require that the speaker names followed by a colon ("ATCo" and "Pilot:") are added, if more than one speaker occurs in an utterance. The above utterance results in the annotation:

ICE68K PILOT STATION REYK_RADAR,
ICE68K PILOT GREETING,
ICE68K PILOT REPORTING ALTITUDE 180 FL,
ICE68K PILOT REPORTING CLIMB 290 none,
ICE68K STATION REYK_RADAR,
ICE68K GREETING,
ICE68K INIT_RESPONSE,
ICE68K CLIMB 360 FL.

We add the *optional* Speaker field in the annotations (turquoise) only if the speaker is not the ATCo. If an altitude report or a clearance contains neither "feet" nor "flight level", the unit field is set to "none". The reason field with value "REPORTING" is used for pilot speakers only, if the altitude value is not a readback and is also not an altitude request. It is, however, not always certain if, e.g., "… descending flight level two five zero" from a pilot is an altitude readback or a report. Both "ICE68L PILOT REPORTING DESCEND 250 FL" and "ICE68L PILOT DESCEND 250 FL" are, therefore, possible.

One could easily determine which one is correct, by looking into the previous utterances. The annotation rules, however, require considering only the current utterance for creating the annotations. This needs to be considered when comparing two different semantic extraction applications. Similarly, when a pilot requests a command from the ATCo, the annotation is associated with a REQUEST value for the reason field. For example, "speed bird four two alfa requesting flight level one two zero" from a pilot would correspond to "BAW42A PILOT REQUEST ALTITUDE 120 FL". It is important to note that the REPORTING value is only allowed for pilot utterances. The value REQUEST normally occurs also only in pilot utterances, put it could also occur during ATCo-ATCo negotiation.

It is important to note that if a callsign is not fully said by the ATCo or is not completely understood, the annotation always contains the full callsign of an aircraft. For example, "easy jet one zero one delta", "easy jet one delta" or "easy one delta" - all forms are represented by callsign EZY101D.

The following very long utterance from enroute airspace "*november triple nine papa november after passing five eight north five zero west reroute direct doryy spelling is delta oscar romeo yankee yankee you can expect further routing by gander control later on*" results in only one command with a condition.

N999PN DIRECT_TO DORYY none
WHEN PASSING 58N_050W

The non-color-highlighted part is just additional information, which is not covered by the ontology. DORYY occurs only once. "none" is used, because no direction qualifier "LEFT" or "RIGHT" is provided. The purple part highlights the condition for the DIRECT_TO-command.

The utterance "*right turn direct five eight north five zero west then reroute direct doryy*" results in a command with two values, but without a condition. Here we have a qualifier "RIGHT":

N999PN DIRECT_TO 58N_050W DORYY RIGHT

The following utterance contains one reporting and two requests from the pilot: "*ice air two seven four climb flight level three seven zero request flight level three nine zero and mach decimal seven nine*".

ICE274 PILOT REPORTING CLIMB 370 FL
ICE274 PILOT REQUEST ALTITUDE 390 FL
ICE274 PILOT REQUEST SPEED 0.79 MA

The word sequence "mach decimal seven nine" in the utterance is somewhat ambiguous. It could also mean just a REPORTING of the current speed value. We decided for the REQUEST, because it follows a request and not a reporting. However, the following answer of the ATCo shows that it is a reporting. Though, the ontology rules are defined as to annotate each utterance independently from other utterances, i.e., without considering additional knowledge from recent utterances and especially from future utterances.

The transcription *"okay we check thanks air canada eight three four"* results in "ACA834 NO_CONCEPT" - not all words are covered by the ontology rules. *"okay we check thanks air canada eight three four descend four thousand feet"* would, however, result in "ACA834 DESCEND 4000 ft". NO_CONCEPT is extracted only if no other command type is extracted for this callsign.

As mentioned earlier, the ontology is continuously expanded and this enables better evaluation of speech recognition performance on a semantic level as more concepts are recognized. An implementation of the ontology from DLR already exists, which includes an automatic extraction (command recognition) of ontology concepts from word sequences. In general, the command extraction first looks for fully matching callsigns followed by extraction of complete commands, incomplete commands (i.e., clearances given without known keywords), and values. The final step extracts non-fully matching callsigns from words which do not belong to an already extracted command. More details are provided in [28].

IV. Metric for Semantic Extraction Accuracy

The user or the ATCo using speech recognition is interested in a high recognition rate and a low error rate on a semantic level. In other words, the meaning behind the spoken word sequence must be interpreted correctly [29]. Quantifying the accuracy on semantic level, i.e., recognition accuracy and error rate, is described in this section. We use command recognition rate, command recognition error rate, and command recognition rejection rate, in order to be consistent with [30]. Nevertheless, a wrong condition counts as an error while computing the command recognition error rate.

A. Basic Example for Metric Calculation

Command recognition rates are computed by comparing instructions from **manual** human annotation (gold annotation) to the results of the **automatic** semantic command extraction (automatic annotation). For a given speech utterance, each instruction (see Figure 1) is treated as one big word. Then, the Levenshtein distance between the gold annotation and the results of command extraction is calculated, resulting in the number of substitutions (subs), insertions (ins), and deletions (del). The Table 1 gives an overview about the different metrics and illustrates an example how they are calculated. In the table #gold defines the total number of commands in the gold annotation. #match defines the number of matches, which is #gold – subs – del. If the result of command extraction contains either NO_CONCEPT or NO_CALLSIGN, these substitutions and insertions are always calculated as deletions, i.e., these extractions contribute to the rejection rate and not to the error rate (as shown in the example in Table 1).

Table 1: Metric Definition

Metric	Calculation
Command Recognition Rate (**RcR**)	**RcR** = #matches / #gold
Command Recognition Error Rate (**ErR**)	**ErR** = (subs + ins) / #gold
Command Recognition Rejection Rate (**RjR**)	**RjR** = del / #gold
Callsign Recognition Rate (**CaR**)	Same as RcR but only for callsigns without instructions
Callsign Recognition Error Rate (**CaE**)	Same as ErR, but only for callsigns without instructions
Callsign Recognition Rejection Rate (**CaRj**)	Same as RjR, but only for callsigns without instructions
Unclassified word rate (UnClWR)	UnClWR = #number of unclassified words / #total number of words

Metric	Calculation	
If the command extraction results in different callsigns, the calculation is done for each callsign. See example below, which also illustrates that the sum of RcR, ErR, and RjR can exceed 100%.		
Example		
Command Extraction	**Gold Annotation**	
AFR123 DIRECT_TO OKG none AFR123 INIT_RESPONSE AFR123 TURN RIGHT AUA1AB NO_CONCEPT DLH123 NO_CONCEPT	AFR123 INIT_RESPONSE AFR123 TURN LEFT AUA1AB SPEED 140 kt DLH123_NO_CONCEPT	
Result:		
RcR = 2/4 = 50% (green)	*ErR = 2 / 4 = 50% (purple)*	*RjR =1/4 = 25% (yellow)*

For calculating the callsign rates CaR, CaE, and CaRj, we just compare callsigns from the gold annotation and from the automatic extraction (see Table 1). For each utterance we consider the callsign only once, except when different callsigns are annotated or extracted. For the example in Table 1 this results in the three annotated and extracted callsigns AFR123, AUA1AB, and DLH123.

B. Metric Calculation with Disabled Command Types

As the ontology is still evolving, the annotations and extractions for different data sets are based on different versions of the ontology. In most cases new ontology versions introduce new command types. The metric calculation has to take this into account so that older data sets can also be reused. If some command types were not considered in the gold annotation or in the extraction (set via a configuration file), these command types are deleted from both the gold annotation and from the automatic extraction. If after the deletions, the set of annotations or extractions for a callsign is empty, the command type NO_CONCEPT is added for this callsign. If INIT_RESPONSE and SPEED command types are not supported for the above example from the metric definition, this would lead to the following result as shown in Table 2.

Table 2: Example of Metric Definition with INIT_RESPONSE and SPEED Commands Disabled

Command Extraction	Gold Annotation	
AFR123 DIRECT_TO OKG none AFR123 TURN RIGHT AUA1AB NO_CONCEPT DLH123 NO_CONCEPT	AFR123 TURN LEFT AUA1AB NO_CONCEPT DLH123 NO_CONCEPT	
AFR123 INIT_RESPONSE is mapped to AFR123 NO_CONCEPT. However, both gold annotation and command extraction still contain another command for AFR123. NO_CONCEPT is only added if it is the only command, which is the case for AUA1AB with SPEED mapped to NO_CONCEPT.		
Result:		
RcR = 2/3 = 67% (green)	*ErR = 2/3 = 67% (purple)*	*RjR = 0 = 0%*

C. Metric Calculation with Additional Command Types

As mentioned earlier, it might be important to extract the number of greetings and farewells in an utterance, because using often greetings and farewells might be an indication for situations with reduced workload for the ATCos. Table 3 shows an example. The gold annotations for a given data set could be

generated years ago. Like a few other newly introduced command types, FAREWELL and GREETING were not supported by the ontology's first versions. Let's assume that it is decided now to also support these two command types by command extraction to enable workload estimation. Gold transcriptions and gold annotations are expensive, because they require manual checking. They are needed as reference for performance evaluation. Therefore, they should be reused, whenever possible for evaluation of different command extraction implementations. The possibility to exclude some command types from evaluation (here GREETING and FAREWELL) is important. Table 3 shows that without this exclusion possibility we would get a command recognition error rate (ErR) of 33% and 0% when excluding GREETING and FAREWELL from evaluation for such datasets.

TABLE 3: EXAMPLE OF METRIC DEFINITION WITH GREETING AND FAREWELL ACTIVATED

Command Extraction	Gold Annotation
AUA7H GREETING AUA7H STATION RADAR AUA7H INIT_RESPONSE AUA7H DESCEND 130 FL AUA7H INFORMATION ATIS L	AUA7H STATION RADAR AUA7H INIT_RESPONSE AUA7H DESCEND 130 FL AUA7H INFORMATION ATIS L
The utterance *"**good evening** austrian seven hotel praha radar radar contact descend flight level one three zero lima correct"* results in the above extraction, if GREETING is supported. But the annotation performed earlier did not contain a GREETING command since they did not exist back then.	
CSA904 CONTACT RADAR *CSA904 CNT_FREQ 127.825* ***CSA904 FAREWELL***	*CSA904 CONTACT RADAR* *CSA904 CNT_FREQ 127.825*
The utterance *"CSA nine zero four contact praha radar one two seven decimal eight two five ahoj"* contains a FAREWELL command type. CONTACT_FREQUENCY is abbreviated as CNT_FREQ.	
Result:	
RcR = 6/6 = 100% (green)	*ErR = 2/6 = 33% (purple)* / *RjR = 0 = 0%*

However, the other way around is also possible wherein the annotations contain more command types than what the command extraction implementations support. In this case, GREETING and FAREWELL would have to be excluded from the gold annotations.

D. Rate of Unclassified Words as Error Indicator

The metric unclassified word rate (UnClWR) is the proportion of words in an utterance which are classified as "unknown". In other words, it is the total number of words which are classified as "unknown" after executing command extraction on a given utterance divided by the total number of words in the utterance. Unclassified word rate is relevant because it is an indication that the command extractor could not recognize and map them to corresponding concepts, thereby pointing to possible errors made by the ASR. The metric could especially help to evaluate the extraction performance on automatically transcribed data or on automatically annotated training data, i.e., data sets for which gold transcription or gold annotations, respectively, are not available.

Table 4 shows an example of good classification of the extraction algorithm. Just one word is mark as unknown ("unkn").

TABLE 4: GOOD CLASSIFICATION PERFORMANCE WITH JUST ONE UNKNOWN

cont*	heading	zero six	zero descend	altitude	six thousand
unkn	type	valu valu valu	type	type valu	valu

The example in Table 5 is a counter example. Most of the words could not be classified. Nevertheless, the command "SPEED 250 none" is still extracted. The classification results from an automatic transcription. The gold transcription is here "you will be following a heavy triple seven speed now two fifty or below".

TABLE 5: BAD CLASSIFICATION PERFORMANCE WITH SEVEN UNKNOWNS

level four	one heavy triple seven	speed	now	two fifty
unkn unkn unkn	unkn unkn unkn	type	unkn	valu valu

V. EXPERIMENTAL RESULTS

A. Comparison of Word Error Rates with Semantic Recognition Rates

Voice and surveillance data from Prague Approach (Czech Republic) and Vienna Approach (Austria) from the two SESAR projects *MALORCA* and *CWP HMI* were used for both Prague and Vienna gold transcriptions and gold annotations of the ATCo voice recordings were available. From simulation runs (Lab) of the *CWP HMI* project 6,885 commands were taken from five different ATCos from Prague and 6,005 commands were taken from two different ATCos from Vienna (see rows with *Labs*) [31]. From the MALORCA project 6,094 commands from Prague approach and 4,417 commands from Vienna approach were taken from operational environment recordings of 12 and 41 ATCos [32], respectively (see Table 6 with rows "*Ops*"). The number of commands per speech utterance was between one and seven.

TABLE 6: RECOGNITION ACCURACY FOR OPS ROOM AND LAB

	#Cmd	#Utt	RcR	ErR	CaR
Ops Prague	6,094	3,038	98.5%	0.9%	99.8%
Lab Prague	6,885	4,211	99.2%	0.5%	99.7%
Ops Vienna	4,417	2,279	94.8%	4.0%	98.2%
Lab Vienna	6,005	3,562	95.3%	2.5%	96.4%

Table 6 shows the metrics, number of commands (#Cmd), and speech utterances (#Utt) for the different data sets. The command extractions in this table are performed on the gold transcriptions (WER=0%) and, therefore, shows the upper limit of command extraction if the word recognition is perfect. More interesting are the results, when the output from a speech-to-text engine with WERs > 0% is used. For the results of Table 7 different models and context information from surveillance and flight plan data were used, resulting in different WERs. We provide only data from the ops room environment. The difference for the lab environment with different speaker models is very minor.

Table 7 shows the results, if four different speech-to-text engines are used (i) manual (human) transcription, (ii) automatic transcription trained with many different speakers, but

not using the information of the available callsigns, (iii) as before, but using the information of available callsigns, (iv) as (iii), but trained only from utterances of just one speaker and nevertheless used to recognize speech also from other speakers (rows with "bad speech model").

TABLE 7: RECOGNITION ACCURACY WITH DIFFERENT WERS

	RcR	CaR	WER
Ops Prague, gold transcription	98.5%	99.8%	0.0%
Ops Prague, no callsign context	96.5%	98.7%	2.3%
Ops Prague, callsign context	96.6%	98.2%	2.8%
Ops Prague, bad speech model	76.8%	88.5%	13.5%
Ops Vienna, gold transcription	94.8%	98.2%	0.0%
Ops Vienna, no callsign context	89.9%	93.0%	5.1%
Ops Vienna, callsign context	88.6%	91.6%	6.7%
Ops Vienna, bad speech model	82.7%	87.8%	9.5%

This results in a bad performance concerning WER. We see that a lower WER of 2.3% results in a worse command recognition rate (96.5%) as compared to a WER of 2.8%. The latter WER is based on using the context information, i.e., information regarding which aircraft callsigns are currently controlled by the ATCo. The gold transcription "austrian two three one" is then recognized as, e.g., "austrian three three one" if only AUA331 is available in context, although the ATCo clearly said "two three". Another interpretation is that, e.g., "ryanair" is automatically transcribed, whereas just "air" was understandable in the utterance.

The data does not only show that a lower WER does not automatically result in a higher command recognition rate, but also shows that fully recognizing an instruction/command does not require each word of the command to be correctly recognized. The command extraction algorithm always uses the information as to which callsigns are currently in the air, independent of the fact whether the speech-to-text block uses this information or not.

Table 8 shows what command recognition rates could be expected for certain WER and different average command length in words, if the WER would directly translate to the command recognition rate provided that the recognition results for a word are independent from the recognition results of the previous words, which is not true.

TABLE 8: COMMAND LENGTH IN NUMBER OF WORDS

WER	3	4	5	6	7	8	9
2.3%	93%	91%	89%	87%	85%	83%	81%
2.8%	92%	89%	87%	84%	82%	80%	78%
5.1%	85%	81%	77%	73%	69%	66%	62%
6.7%	81%	76%	71%	66%	62%	57%	54%
9.5%	74%	67%	61%	55%	50%	45%	41%
13.6%	65%	56%	48%	42%	36%	31%	27%

Assuming that the sequence of words "descend flight level two one zero" consisting of six words only results in "DESCEND 120 FL" if all six words are correctly recognized, should result in a command recognition rate of 55% given a WER of 9.1%. The average command length for Prague and Vienna data were 7.0 and 5.6 words, respectively.

So, for a WER of 2.8% a command recognition rate of at most 82% should result, but we have achieved 96.6% (as shown in Table 7). Similarly, for a WER of 5.1% for Vienna ops room data without using callsign information from the surveillance data, we expect a command recognition rate of about 75%, but we observed 89.9% recognition rate. The command extraction algorithm is quite robust, which was also shown by Ohneiser et al. on tower utterances from Lithuanian ATCos [33].

Table 9 illustrates the results, if we concentrate on altitude changing command types (column *DESCEND*) and direction changing command types (column *DIRECT_TO*), which are important in the ATC world. The top part of Table 9 shows the results for Prague and the bottom part for Vienna ops room data. The command recognition rate RcR for the DESCEND command decreases only slightly with increasing WER within *acceptable* levels, i.e., it decreases by less than 3% absolute when WER is below 7%. In these cases, RcR for the DESCEND command is better than RcR for all commands shown in Table 9.

TABLE 9: SPECIFIC COMMAND RECOGNITION RATES

Ops Prague			
	All	**DESCEND**	**DIRECT_TO**
WER	**6063**	**925**	**370**
0.0%	98.5%	99.8%	97.0%
2.3%	96.5%	98.3%	95.1%
2.8%	96.6%	99.0%	87.8%
13.6%	76.8%	76.1%	77.3%
Ops Vienna			
	All	**DESCEND**	**DIRECT_TO**
WER	**4417**	**679**	**387**
0.0%	94.8%	98.5%	91.0%
5.1%	89.9%	95.9%	86.6%
6.7%	88.6%	95.4%	82.2%
9.5%	82.7%	86.5%	77.3%

However, the command recognition rate decreases significantly for DESCEND command if WER gets worse, i.e., worse than 9%. For such cases, the command recognition rate for DESCEND command is not better than the overall command recognition rate averaged over all command types. The performance of RcR for the DIRECT_TO command, however, decreases already, when the WER gets slightly worse.

B. Recognition Rates considering Additional Command Types

The recordings from Vienna and Prague were annotated in 2017 during the MALORCA project and in 2019. At that time GREETING and FAREWELL were not annotated, i.e., all the results reported in Table 7 and Table 9 do not consider these command types, although the command extraction implementation supports them. These extractions are, however, transformed to "NO_CONCEPT", before evaluation starts.

Table 10 shows the command recognition error rates (ErR) for the cases when GREETING and FAREWELL commands are ignored and without ignoring them. The error rate dramatically increases from 2.0% when these commands are ignored to about 12.7% when they are not ignored.

The table also shows the difference between a simulation in the labs and real-life utterances from the ops room. FAREWELL is seldom used in the lab environment, whereas greetings often occur in the ops room. For DESCEND and CONTACT command types, however no big difference in their frequency of occurrence is observed.

TABLE 10: EFFECT OF IGNORING GREETING AND FAREWELL ON ERR

	Lab	Ops
Total number of commands	6885	6094
Number of GREETING commands	83	488
Number of FAREWELL commands	2	201
Number of DESCEND commands	1390	925
Number of CONTACT commands	569	522
ErR, switching off FAREWELL, GREETING	0.5%	2.0%
ErR, switching on FAREWELL, GREETING	1.7%	12.7%

C. *Recognition Rates considering Unclassified Words*

Table 11 illustrates and compares the results of command extraction for gold and automatic transcription by also considering the number of unclassified words. The results are extracted from a data set from the London terminal maneuvering area (TMA). We concentrate on pilot utterances because of increased noise and reduced audio quality for pilots, thereby resulting in decreased ASR performance for pilots with respect to WER as compared to ATCos. Therefore, the relationship between unclassified words and the recognition rate can be better illustrated for pilot utterances.

TABLE 11: CORRELATION BETWEEN UNCLASSIFIED WORD RATE (UNCLWR) AND RECOGNITION RATE (RCR)

Dataset	RcR (gold)	UnClWR (gold)	RcR (automatic)	UnClWR (automatic)	WER (automatic)
Dir1	94.1%	6.9%	84.2%	8.9%	6.6%
Dir2	92.3%	9.0%	69.1%	12.4%	11.8%
Dir3	94.6%	8.6%	86.6%	9.0%	5.3%
Dir4	94.4%	9.6%	89.1%	10.1%	5.2%

From Table 11, we see that the unclassified word rate (UnClWR) increases from gold to automatic transcriptions, the command recognition rates decrease. For example, for Dir1 UnClWR for gold and automatic transcriptions are 6.9% and 8.9%, respectively. The command recognition rate decreases from 94.1% to 84.2% when using automatic transcriptions. This is an indication of the presence of errors in the automatic transcription and is reflected in the WER. The higher the WER, the more words will remain unrecognized, resulting in lower command recognition rates.

For instance, the WER of Dir2 is relatively higher at about 11.8%. This leads to a higher increase in UnClWR from 9% to 12.4%, thereby significantly reducing the recognition rate from 92.3% to 69.1%. This shows that there is a strong negative correlation between the command recognition rate and the unclassified word rate for pilots. The correlation coefficient is -0.85. On the other hand, there is a positive correlation between the word error rate and the unclassified word rate, which is about 0.78. Since there is a positive correlation between the word error rate and unclassified word rate, the presence of higher number of unclassified words in an utterance could be used as a hint that the command extraction is wrong. This is heuristic to reduce the command recognition error rate. Further analysis on more data sets is necessary in order to get statistically significant results. Here we could only show a trend.

For ATCo utterances, there is no correlation and this is reflected in the relatively low correlation coefficients of 0.42 and -0.47 for recognition rates and the errors rates, respectively. This could be attributed to the lower WERs for ATCo utterances.

Helmke et al. showed for the application of readback analysis that there is a significant dependence between the rate of unclassified words and the recording environment [34]. In lab environment a UnClWR on gold transcriptions of 1.2% for Prague and of 4.3% for Vienna, respectively, was observed, whereas in ops environment 10% for Prague and 12% for Vienna were observed.

VI. CONCLUSIONS

The paper has extended the ontology developed by SESAR solution CWP HMI also for pilot utterances. The implementation of the ontology rules results in command recognition rates of 99% for Prague airport and achieves 95% for Vienna airport, when manually transcribed utterances are used.

The implementation is robust against errors resulting from speech-to-text transformation. WER below 3% decreases performance of command recognition rate only slightly. WER above 10% still enable command recognition rates above 75%, even though the average command length was longer than 6 words. Command extraction from automatically transcribed data with WER of 3% for Prague or 6% for Vienna achieves 96% for Prague and 88% for Vienna, respectively. For Vienna the gold annotations are still improvable and the used phraseology contains a high variability often deviating from published standard phraseology [2].

While the command recognition rate metric is not new, the presented ontology for transforming ATC utterances consisting of a sequence of words into its semantic elements, is new. Only the presented definition and the implementation of the extended ontology, enable a detailed comparison of different speech recognition and understanding applications on a semantic level and not just on word level.

Using just the word error rate would represent only half of the truth. However, WER analyses do provide initial hints with respect to the ASR performance. New, however, is the proposed metric of the unclassified word rate, which also enables to evaluate the semantic extraction performance on unlabeled, i.e., untranscribed, ATC utterances.

The results also show that evaluating speech recognition in the lab environment can result in different results compared to ops room environment. If the target environment is the ops room, the evaluation in the lab can only give first hints, but the command recognition rates and command recognition error rates can be very different in the ops room later on.

ACKNOWLEDGEMENTS

Three SESAR2020 industrial research projects PJ.16-04-W1 (CWP HMI), PJ.10-96-W2 (HMI Interaction modes for Approach control), and PJ.05-97-W2 (HMI Interaction Modes for Airport Tower) as well as the exploratory research project HAAWAII have received funding from the SESAR Joint Undertaking under the European Union's grant agreement No. 734141, 874464, 874470, and 884287.

REFERENCES

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

[2] ICAO, "Doc 4444, Procedures for Air Navigation Services, Air Traffic Management," ICAO, Montréal, Canada, 2016.

[3] Said, M. Guillemette, J. Gillespie, C. Couchman, and R. Stilwell, "Pilots & Air Traffic Control Phraseology Study," in International Air Transport Association, 2011.

[4] J.M. Cordero, M. Dorado, and J.M. De Pablo, "Automated speech recognition in ATC environment," in Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems (ATACCS '12). IRIT Press, Toulouse, France, 2012, pp. 46-53.

[5] V.I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," Soviet Physics—Doklady 10.8, Feb. 1966.

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

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

[8] 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," IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), London, United Kingdom, 2018.

[9] J. Godfrey, "The Air Traffic Control Corpus (ATC0) LDC94S14A," 1994. Online available: https://catalog.ldc.upenn.edu/LDC94S14A.

[10] K. Hofbauer and St. Petrik, "ATCOSIM Air Traffic Control Simulation Speech Corpus," Technical Report, May 2008, TR TUG-SPSC-2007-11.

[11] V.N. Nguyen and H. Holone, "N-best list re-ranking using syntactic score: A solution for improving speech recognition accuracy in Air Traffic Control," 16th Int. Conf. on Control, Automation and Systems (ICCAS 2016), Gyeongju, Korea, 2016, pp. 1309–1314.

[12] V.N. Nguyen and H. Holone, "N-best list re-ranking using syntactic relatedness and syntactic score: An approach for improving speech recognition accuracy in Air Traffic Control," 16th Int. Conf. on Control, Automation and Systems (ICCAS 2016), Gyeongju, Korea, 2016, pp. 1315–1319.

[13] D.R. Johnson, V.I. Nenov, and G. Espinoza, "Automatic speech semantic recognition and verification in Air Traffic Control," IEEE/AIAA, 32rd Digital Avionics Systems Conference (DASC), East Syracuse, NY, USA, 2016.

[14] http://www.json.org, JSON = JavaScript Object Notation, n.d.

[15] AcListant homepage: www.AcListant.de, AcListant = Active Listening Assistant, n.d.

[16] A. Schmidt, "Integrating situational context information into an online ASR system for Air Traffic Control," Master Thesis, Saarland University (UdS), 2014.

[17] 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.

[18] MALORCA homepage: www.malorca-project.de, MALORCA = Machine Learning of Recognition Models for Controller Assistance, n.d.

[19] A. Srinivasamurthy, P. Motlicek, I. Himawan, G. Szaszák, Y. Oualil, and H. Helmke, "Semisupervised learning with semantic knowledge extraction for improved speech recognition in air traffic control," INTERSPEECH 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden Aug. 2017.

[20] D. Randall, "Direct Voice Input (DVI) Technology readiness and status introduction," Whitely, Fareham, United Kingdom, 2006.

[21] STARFiSH, research project funded by the German Federal Ministry of Education and Research, see for further information https://www.softwaresysteme.pt-dlr.de/de/ki-in-der-praxis.php, in German, n.d.

[22] PJ.05-97-W2 SESAR2020 funded industrial research projects under the European Union's grant agreement 874464, see for further information https://www.remote-tower.eu/wp/?page_id=888, and https://www.remote-tower.eu/wp/?p=824 and https://www.sesarju.eu/index. php/projects/DTT, n.d.

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

[24] O. Ohneiser, H. Helmke, S. Shetty, M. Kleinert, H. Ehr, Š. Murauskas, T. Pagirys, "Prediction and extraction of tower controller commands for speech recognition applications," Journal of Air Transport Management, Volume 95, 2021, 102089, ISSN 0969-6997.

[25] S. Chen, H.D. Kopald, R. Chong, Y. Wei, and Z. Levonian, "Read back error detection using automatic speech recognition," 12th USA/Europe Air Traffic Management Research and Development Seminar (ATM2017), Seattle, Washington, USA, 2017.

[26] PJ.10-96-W2: SESAR2020 funded industrial research projects under the European Union's grant agreement 874470, https://cordis.europa.eu/programme/id/H2020_SESAR-IR-VLD-WAVE2-10-2019/de, n.d.

[27] HAAWAII homepage: www.haawaii-project.de, Highly Automatic Air Traffic Controller Workstation with Artificial Intelligence Integration, n.d.

[28] H. Helmke, M. Kleinert, O. Ohneiser, H. Ehr, S. Shetty, "Machine Learning of Air Traffic Controller Command Extraction Models for Speech Recognition Applications," IEEE/AIAA 39th Digital Avionics Systems Conference (DASC), virtual conference, 2020.

[29] Y. Lin, "Spoken Instruction Understanding in Air Traffic Control: Challenge, Technique, and Application," Aerospace, 8, No. 3: 65, 2021.

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

[31] 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," 8th SESAR Innovation Days, Salzburg, Austria, 2018.

[32] 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," 9th SESAR Innovation Days, Athens, Greece, 2019.

[33] O. Ohneiser, S. Sarfjoo, H. Helmke, S. Shetty, P. Motlicek, M. Kleinert, H. Ehr, Š. Murauskas, "Robust Command Recognition for Lithuanian Air Traffic Control Tower Utterances," Interspeech, Brno, Czechia, 2021.

[34] 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, J. Harfmann, C. Windisch, "Readback Error Detection by Automatic Speech Recognition to Increase ATM Safety," 13th USA/Europe Air Traffic Management Research and Development Seminar (ATM2021), virtual conference, New Orleans, LA, USA, 2021.