Interspeech 2018
2-6 September 2018, Hyderabad

Iterative Learning of Speech Recognition Models for Air Traffic Control

Ajay Srinivasamurthy[1] *Petr Motlicek*[1] *Mittul Singh*[2] *Youssef Oualil*[2]
Matthias Kleinert[3] *Heiko Ehr*[3] *Hartmut Helmke*[3]

[1]Idiap Research Institute, Martigny, Switzerland
[2]Saarland Informatics Campus, Saarbrücken, Germany
[3]German Aerospace Center, Braunschweig, Germany

Abstract

Automatic Speech Recognition (ASR) has recently proved to be a useful tool to reduce the workload of air traffic controllers leading to significant gains in operational efficiency. Air Traffic Control (ATC) systems in operation rooms around the world generate large amounts of untranscribed speech and radar data each day, which can be utilized to build and improve ASR models. In this paper, we propose an iterative approach that utilizes increasing amounts of untranscribed data to incrementally build the necessary ASR models for an ATC operational area. Our approach uses a semi-supervised learning framework to combine speech and radar data to iteratively update the acoustic model, language model and command prediction model (i.e. prediction of possible commands from radar data for a given air traffic situation) of an ASR system. Starting with seed models built with a limited amount of manually transcribed data, we simulate an operational scenario to adapt and improve the models through semi-supervised learning. Experiments on two independent ATC areas (Vienna and Prague) demonstrate the utility of our proposed methodology that can scale to operational environments with minimal manual effort for learning and adaptation.

Index Terms: Speech recognition, Iterative learning, Semi-supervised learning, Air traffic control

1. Introduction

Automatic Speech Recognition (ASR) is making inroads into our everyday lives through its adoption in several human computer interaction systems, replacing the traditional forms of interaction with natural language interaction. Virtual assistants capable of natural language understanding, even within a limited domain of comprehension, can greatly ease human interaction with machines. One domain that can benefit from such speech technologies is Air Traffic Control (ATC), where voice communication is still the main form of communication with air traffic controllers guiding and navigating aircraft in an airspace through voice commands to pilots.

Currently, the commands issued by controllers are also manually entered and recorded for the purposes of planning and safety, thus doubling effort and adversely affecting controller's productivity. Recently, it has been shown that introducing automation into this process in the form of ASR, called Assistant Based Speech Recognition (ABSR), has the potential to reduce controller's workload and improve operational efficiency [1, 2]. The task of an ABSR system is to recognize a controller's spoken utterances to extract meaningful ATC command hypotheses (text hypotheses are only an intermediate step).

For optimal performance, an ABSR system needs to be adapted within each airspace, which is a time and resource intensive process. Within the MALORCA project (`http://www.malorca-project.de`), we wish to adapt ABSR systems to an airspace and controller with minimum manual intervention. Although ATC communication is limited in vocabulary and is assumed to follow a standard phraseology, it is considered as a challenging task due to low quality speech, multiple English accents, high rate of speaking and local variations/deviations in phraseology. Nevertheless, the commands issued by controllers are accompanied by time-synchronized radar data that can be used to provide a situational context for all commands issued by controllers. ATC systems operate continuously and hence generate increasing amounts of (untranscribed) speech and radar data, motivating us to explore the use of semi-supervised learning methods for building and adapting the corresponding ABSR system. Further, we operate in a bimodal (speech and radar) setting where radar data complements speech to either correct ASR hypotheses or to select data for semi-supervised learning.

ASR in ATC domain has been explored to a limited extent [3, 4] and multi-modal speech recognition has been largely explored with visual data [5]. Methods that utilize the radar data to improve ASR through semi-supervised learning have only been recently explored [2, 6]. In this paper, we propose an iterative semi-supervised learning approach to build and adapt an ABSR system in a bi-modal setting that is representative of the operational environment. The proposed approach can be deployed to build a continuously adapting ABSR system for a new ATC area starting from little (or no) transcribed speech data from the area. We first describe the components of an ABSR system along with the approach we take for iterative learning. We then demonstrate our approach on operational data from two ATC areas - Prague and Vienna.

2. ABSR components

The ABSR system and components are built independently for Vienna and Prague ATC areas. A brief description of each component is presented below.

2.1. Datasets

The speech and radar data used in this paper are described in Table 1 and come from operational environments in two different ATC areas - Prague and Vienna. The data from both ATC areas was recorded in the second half of 2016. The speech was recorded at a sampling rate of 8 kHz and has been seg-

The work in the paper was conducted as a part of MALORCA project, funded by the SESAR Joint Undertaking, under EU Horizon 2020 Research and Innovation programme. The authors thank Air Navigation Services of Czech Republic and AustroControl from Austria for providing data from Prague and Vienna approach, respectively. Youssef Oualil contributed to this paper during his work period at Saarland University. [1]Ajay Srinivasamurthy is currently with Amazon.com, Bangalore, India and contributed to this paper during his work period at Idiap Research Institute before joining Amazon.

10.21437/Interspeech.2018-1447

Dataset	Prague		Vienna	
	Dur.	#Spk.	Dur.	#Spk.
`Train`	3.3	9	2.6	15
`Untrans`	18.3	11	18.2	41
`Test`	1.4	3	1.1	5
Total	23.0	12	21.9	45

Table 1: *Prague and Vienna datasets, showing the duration (Dur. in hours) and number of speakers (#Spk.) in each dataset*

mented into short utterances containing only a few controller commands. The utterances are timestamped and the accompanying radar data is synchronized with them. The pilot replies to controller utterances are not recorded and stored since they are not relevant in our studies. All recordings are assigned with speaker labels but only a part of the dataset is annotated with text and command transcripts using an in-house annotation tool. The speech data comprises different speakers, accents and airspace situations. While the data is not publicly available currently, the speech content of the dataset is similar to other publicly available ATC domain datasets such as the LDC ATC dataset [7] and ATCOSIM dataset [8].

A large part of the data from Prague and Vienna shown in Table 1 is untranscribed (denoted as `Untrans`). The transcribed data is divided into `Train` and `Test` datasets. The `Train` dataset is used to build seed ASR models while `Test` dataset is used to evaluate our proposed approach. The `Test` dataset does not share any speakers with the `Train` dataset. The `Untrans` dataset is untranscribed without any accompanying text/command transcripts and is used to apply iterative semi-supervised learning.

Since the amount of transcribed data available is limited, we also utilize other transcribed resources to train the ASR models. We pool 150 hours of English speech data from the publicly available LIBRISPEECH [9], ICSI [10], AMI [11] and TED-LIUM [12] datasets, which have been extensively used for recognition of conversational speech. The speech data and accompanying transcripts (called `MEGA`) are used in conjunction with `Train` dataset [6].

2.2. Lexicon and Acoustic Model

All possible in-domain words from Vienna and Prague ATC areas such as airlines and waypoints are combined and added to the standard CMU-Sphinx dictionary to form an extended pronunciation lexicon subsequently used with acoustic and language models in ASR engine. There are hence no out of vocabulary words in training or testing.

As an Acoustic Model (AM), we employ conventional DNN/HMM (Deep Neural Network Hidden Markov Model), similar to our preliminary studies [6]. More specifically, to increase amount of labeled data for training, we add the `MEGA` dataset to the limited Prague/Vienna `Train` dataset and used to train a DNN/HMM acoustic model, called as DNN-BASE. Next, we adapt this model to Vienna/Prague ATC area using the corresponding `Train` dataset. The adaptation process first re-initializes and randomizes the weights of the last layer of the previously trained DNN-BASE, keeping the architecture and weights of the other layers unchanged. We finally retrain the entire network using `Train` dataset to obtain supervised-adapted DNN (denoted to as DNN-SUP). This way of reinitializing the last layer and retraining the complete network was found to be effective for supervised adaptation using in-domain data [13, 14, 15].

2.3. Language Model

The standardized phraseology used in ATC suggests the use of a rule based Context-Free Grammar (CFG) that models the controller phraseology to build a language model (LM) [16, 17]. However, a closer analysis reveals that controllers often deviate from standards in operational environments and a statistical LM can model such deviations more effectively. In this paper, we use a hybrid approach that combines a grammar-induced class language model with a conventional n-gram language model.

To induce classes, we use the grammar to tag command words with ATC concepts and use the concepts as the word classes. However, there are out-of-grammar words due to the variations introduced by controllers that cannot be assigned to a class, in which case the word itself is identified as the class. We then approximate these grammar-induced class LM to an n-gram LM using a variant of probability-conversion method [18] described in [19], where a given n-gram set is scored with the class-based model and the obtained scores are converted into a back-off n-gram model.

We first build a 3-gram language model with Kneser-Ney (KN3) smoothing [20] from the training text data (data-LM). Sampling large amounts of text from language models can help obtain a good coverage of possible events in the test set. Hence, we complement the limited transcribed text data with additional text by sampling the data-LM. The sampled text is then scored with the above-described class-based LM and converted to an approximated 3-gram LM. Finally, the class-based approximated 3-gram LM is interpolated with the data-LM and then applied to the FST-based decoder. DNN-SUP acoustic model along with this interpolated language model for the supervised-adapted baseline ASR system denoted as ASR-SA.

2.4. Concept Extraction

ASR system outputs word sequences from which we extract the concepts and commands represented by this sequence. Concepts include all meaningful words or expressions which are related to the controller's command and the required action of the aircraft. Concepts mainly include (i) the callsign composed of an airline identifier (International Civil Aviation Organization airline code) and a flight number, (ii) the command word or expression itself, and (iii) the command attributes (usually target values for some flight parameters). This sequence of concepts forms a command. For example, the following utterance "*hello air france six echo tango descend to flight level one six zero*" contains the following concepts:

- AFR6ET (*air france six echo tango* - callsign)
- DESCEND (*descend* - command word)
- 160 (*one six zero* - flight level attribute).

The complete command is hence *AFR6ET DESCEND 160.* To extract concepts from the utterance, we use the rule based CFG used to build the LM (Section 2.3). Each semantic slot for the command is tagged in the CFG, and hence transducing [21] a text hypothesis from the ASR over the CFG results in a semantically tagged version of the text transcript, which is then formatted into a command. If transductions fail (due to a deviation in phraseology not modelled by the CFG or due to ASR errors), the command extractor returns "NO_CALLSIGN" if the callsign is missed, and "NO_CONCEPT", if the command word or the command attribute could not be recovered. Thus, using the AM, LM and the concept extractor, given a speech utterance by a controller, we obtain a plain text hypothesis (sequence of words as they were spoken) and the command hypothesis (intended semantic command).

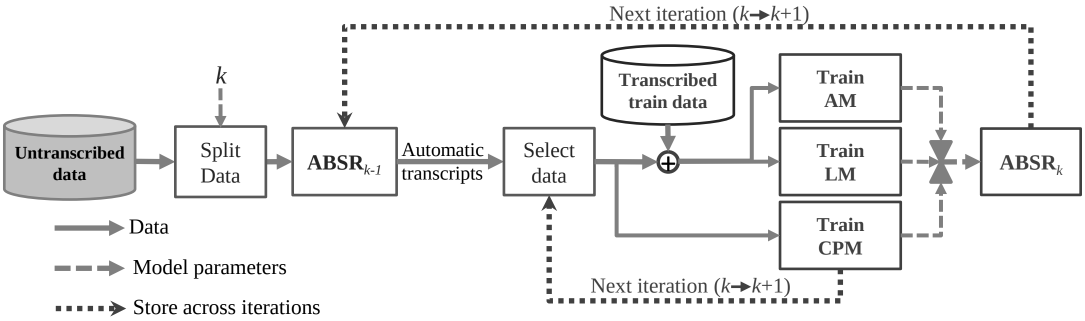

Figure 1: *Iterative learning of an ABSR system. The iterations $k = \{1, \cdots, K\}$ correspond to using 25%, 50%, 75% and 100% of available untranscribed data, respectively. $k = 0$ corresponds to using only transcribed training data (0% untranscribed data).*

2.5. Command Prediction Model

The radar data accompanying speech data provides dynamically changing snapshots (i.e. every few seconds) of the airspace and hence can be used to provide a situational context for the ASR engine. Given a dataset of airspace situations (encoded in the radar data) and the corresponding ground-truth commands issued by controllers, we build a Command Prediction Model (CPM) [22, 23] that provides a list of plausible commands for a given airspace situation. This list of commands is called the dynamic context and is used for iterative semi-supervised learning. The context can be used to reduce the search space of the ASR decoder during recognition or be used to correct generated ASR hypotheses. In this paper, we use the context for both command recognition and correction, as we describe in the next section.

3. Iterative Learning

With the components of the ABSR system described in Section 2 - AM, LM and CPM, we propose an iterative learning approach to build and adapt the system to a new ATC area. A block diagram of the proposed approach is shown in Figure 1. We simulate the increasing amounts of untranscribed data in operational environments by chronologically arranging the controller sessions in `Untrans` dataset and splitting it into a sequence of K data splits $\{S_1, S_2, \cdots, S_K\}$, such that S_k is a subset of the next S_{k+1}. S_0 contains no data from `Untrans` dataset while S_K refers to the complete `Untrans` dataset. In this paper, we consider $K = 4$ data splits, with S_k containing $k \cdot 25\%$ of untranscribed data.

The proposed iterative learning method has three main steps: (1) we generate the automatic transcripts of the current data split using the current ASR model, (2) use the current CPM to select data from those automatic transcriptions and (3) combine the selected data with `Train` data to adapt AM, LM and CPM. We describe each step further in detail.

3.1. Automatic transcript generation

We generate the automatic text and command transcripts for the S_k subset of `Untrans` dataset using the ABSR system from iteration $k-1$. N-best text hypotheses are generated for each utterance and the best text and command hypothesis is estimated as the hypothesis that has the lowest Levenshtein distance to the set of plausible commands predicted by the CPM. This enables us to correct the output hypothesis based on situational context information. For evaluation, the text and command hypothesis for the `Test` dataset is also generated in the same way.

3.2. Data selection

The aim of data selection is to augment the `Train` dataset by automatically transcribing `Untrans` data and selecting the best possible transcripts we can rely on. We use the complementary information from the radar data and utilize the situational context to select or reject a text/command hypothesis and the corresponding audio recording. If the automatically generated command transcript is plausible under the situational context predicted by the CPM, it is selected and retained to augment training data, else it is rejected. This is based on an assumption that if the ASR output text/command hypothesis is also plausible from the perspective of radar, it is likely to be correct. In addition, utterances for which the output is NO_CALLSIGN and/or NO_CONCEPT are also rejected since that indicates an error either in the ASR or concept extraction. The automatically transcribed data from S_k thus selected is denoted as $S_k^\star$ and is combined with `Train` dataset for further model retraining.

3.3. Model retraining

The combined data ($S_k^\star$ + `Train` dataset) is used to adapt and retrain the AM and LM. We adopt the same method as described in Section 2.2 to adapt the AM using the combined data, starting from DNN-BASE. The LM is retrained with the combined data transcripts to rebuild data-LM and then combining it with the the class-based approximated 3-gram LM as described in Section 2.3. In this paper, CPM training uses only untranscribed data and hence only $S_k^\star$ subset is used for retraining.

The models trained from $S_k^\star$ and `Train` dataset form the ASR$_k$ system, which is then used for automatic transcription in the next iteration $k + 1$. It is important to note that the pivot iteration $k = 0$ starts with the supervised-adapted baseline ASR-SA system. Further, the CPM is trained only using automatically transcribed data, and since $k = 0$ iteration comprises only `Train` data, a trained CPM is not available to correct text and command hypotheses. Hence for $k = 0$, a 1-best text and command hypothesis is generated without any correction by CPM, instead of N-best hypotheses. We finally note that this process of incremental iterative semi-supervised learning is suitable for this task since models can be updated often and at regular intervals as more data is available through continuous operation of ATC systems.

4. Experiments

The iterative learning experiments are conducted separately for Prague and Vienna ATC areas. The AM and LM were built using Kaldi [24]. We conduct four iterations of model retraining with splits of 25%, 50%, 75% and 100% of `Untrans` dataset.

As input features to the DNN/HMM acoustic model, we

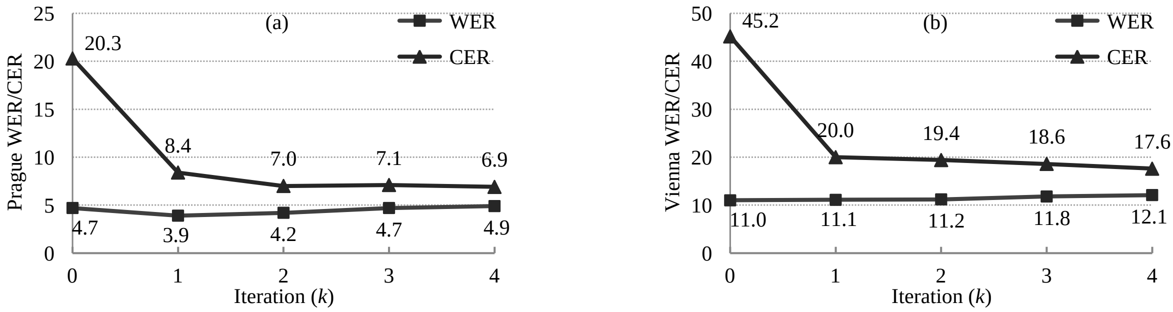

Figure 2: *WER and CER measures for iterative semi-supervised learning over different data split iterations for both (a) Prague and (b) Vienna ATC areas. The numbers on the plot show the value of WER or CER. Similar to Figure 1, iteration $k = 0$ refers to ASR-SA system, and $k = \{1, \cdots, K\}$ correspond to using 25%, 50%, 75% and 100% of total untranscribed data, respectively.*

Split (S_k)	Prague Total (hours)	 Select. (%)	Vienna Total (hours)	 Select. (%)
S_1 (25%)	4.5	56%	4.6	49%
S_2 (50%)	9.2	76%	9.1	74%
S_3 (75%)	13.7	78%	13.7	77%
S_4 (100%)	18.3	78%	18.2	79%

Table 2: *Amount of total untranscribed data S_k (Total, in hours) and selected data $S_k^\star$ (expressed as a % of S_k) from the* `Untrans` *dataset over iterations (k). S_4(100%) corresponds to the complete* `Untrans` *dataset.*

applied 13 dim MFCCs with their delta and acceleration coefficients (39 dim feature vector), along with fMLLR transforms for speaker adaptive training. The DNN comprises 4 layers: 351 dim input layer (9 stacked feature vectors with a context of 4 frames around the centered frame), hidden layers of 1200 nodes and output layer with 3800 units modeling senones. The DNN is trained to minimize frame-level cross entropy. For decoding, except for $k = 0$ iteration when there is no available CPM, we generate 5-best hypotheses and then correct them to the 1-best hypothesis using the situational context.

The most relevant metric of performance for ATC applications is at the command level. However, since the ASR system outputs hypotheses at both word level and command level, we report the commonly used Word Error Rate (WER) and the Command Error Rate (CER). The CER is interpreted with the whole command as one unit, comparing the ground truth command with the hypothesized command. Even a single misrecognized word can cause the whole command to be wrong and hence CER is a much stricter measure of evaluation. We use the supervised-adapted ASR-SA system (corresponding to $k = 0$) as the baseline system in evaluation. We aim to analyze the effect of increasing amounts of data in such an iterative learning task. We report results of evaluating the ASR systems with both WER and CER measures on the Prague and Vienna `Test` datasets. In addition, we also report the results of data selection to evaluate and analyze the role of CPM in data selection for semi-supervised learning.

4.1. Results and Discussion

Table 2 reports the total amount of untranscribed data available in each data split S_k and the percentage of that data selected ($S_k^\star$) to augment `Train` dataset for both Prague and Vienna ATC areas. From the table, we see that a similar fraction of the data is selected during each iteration in both Prague and Vienna, indicating that the data selection method generalizes over these datasets. We see that the fraction of data selected with $k = 1$ iteration is low for both areas since automatic transcripts are generated from 1-best hypothesis without correction by a CPM, hence with higher errors. When a CPM is used, more than around three quarters of data is selected for retraining, implying that the context information is useful for correcting output hypothesis and for data selection.

The performance of iterative semi-supervised learning is summarized in Figure 2. The WER and CER presented are computed over the unseen `Test` dataset of Vienna and Prague. In general, from the table, we observe that CER is much higher than WER for both areas since it is a stricter measure. We also see that the WER and CER on Vienna data is higher than that for Prague, attributed to the noisier data from Vienna compared to Prague. There is a significant reduction in WER and CER using $k = 1$ (25% data split) compared to $k = 0$ that does not use any CPM. This reduction seen shows that 5-best hypotheses and the CPM are useful to correct and improve performance of the system. Despite the observation that WER seems to increase through the next iterations for both Prague and Vienna areas, we see that iterative learning reduces CER through the iterations.

The goal of the application and that of iterative learning is to reduce CER incrementally. We observe a relative decrease in CER of 17.8% and 12% with Prague and Vienna, respectively from k = 1 (25% split) to k = 4 (using 100% untranscribed data) while the WER slightly degrades. The reason behind this is that many components in the ABSR system, such as the CPM [22] and the concept extractor [25] use approaches that were designed to optimize the CER (i.e., the primary metric for ABSR) which does not necessarily optimize the WER. A detailed analysis and discussion of this difference can be found in [25].

5. Conclusions

We proposed an iterative semi-supervised approach to build and adapt an ABSR system to a new ATC area by adapting the AM, LM and CPM with limited transcribed data and increasing amounts of untranscribed data, which is reflective of operational environments. We exploited the bi-modal nature of the problem and developed a radar data based data selection method for untranscribed data. Our experiments on data from Prague and Vienna ATC areas show a significant improvement over a baseline that does not use any untranscribed data with further improvements in CER in subsequent iterations despite a marginal increase in WER. We built and evaluated systems for Prague and Vienna separately. In the future, we wish to explore combining all data sources and do cross domain adaptation for different ATC areas.

6. References

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

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

[3] H. D. Kopald, A. Chanen, S. Chen, E. C. Smith, and R. M. Tarakan, "Applying automatic speech recognition technology to air traffic management," in *Proc. of the IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*, 2013.

[4] D. Schäfer, "Context-sensitive speech recognition in the air traffic control simulation," Ph.D. dissertation, University of Armed Forces, Munich, 2001.

[5] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proceedings of the IEEE*, vol. 91, no. 9, pp. 1306–1326, Sep. 2003.

[6] A. Srinivasamurthy, P. Motlicek, I. Himawan, G. Szaszák, Y. Oualil, and H. Helmke, "Semi-Supervised Learning with Semantic Knowledge Extraction for Improved Speech Recognition in Air Traffic Control," in *Proc. of INTERSPEECH*, Stockholm, Sweden, Aug. 2017, pp. 2406–2410.

[7] J. Godfrey, "Air Traffic Control Complete LDC94S14A," DVD, 1994. [Online]. Available: http://catalog.ldc.upenn.edu/LDC94S14A

[8] K. Hofbauer, S. Petrik, and H. Hering, "The ATCOSIM Corpus of Non-Prompted Clean Air Traffic Control Speech," in *Proc. of the 6th International Conference on Language Resources and Evaluation (LREC)*, 2008, pp. 2147–2152.

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

[10] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI meeting corpus," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003, pp. I–364–367.

[11] J. Carletta, "Announcing the AMI meeting corpus," *The ELRA Newsletter*, vol. 11, no. 1, pp. 3–5, 2006.

[12] A. Rousseau, P. Deléglise, and Y. Estève, "Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks," in *Proc. of the 9th International Conference on Language Resources and Evaluation (LREC)*, 2014, pp. 3935–3939.

[13] A. Lazaridis, I. Himawan, P. Motlicek, I. Mporas, and P. N. Garner, "Investigating cross-lingual multi-level adaptive networks: The importance of the correlation of source and target languages," in *Proceedings of the International Workshop on Spoken Language Translation*, Dec. 2016.

[14] D. Imseng, P. Motlicek, P. N. Garner, and H. Bourlard, "Impact of deep mlp architecture on different acoustic modeling techniques for under-resourced speech recognition," in *Proc. of the IEEE workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, Dec. 2013.

[15] I. Himawan, P. Motlicek, D. Imseng, B. Potard, N. Kim, and J. Lee, "Learning Feature Mapping using Deep Neural Network Bottleneck Features for Distant Large Vocabulary Speech Recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Brisbane, Australia, Apr. 2015, pp. 4540–4544.

[16] Y. Oualil, M. Schulder, H. Helmke, A. Schmidt, and D. Klakow, "Real-time integration of dynamic context information for improving automatic speech recognition," in *Proc. of INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2107–2111.

[17] G. Lecorvé, J. Dines, T. Hain, and P. Motlicek, "Supervised and unsupervised Web-based language model domain adaptation," in *Proc. of INTERSPEECH 2012*, Sep. 2012.

[18] H. Adel, K. Kirchhoff, N. T. Vu, D. Telaar, and T. Schultz, "Comparing approaches to convert recurrent neural networks into backoff language models for efficient decoding," in *Proc. of INTERSPEECH 2014*, Singapore, Sep. 2014, pp. 651–655.

[19] M. Singh, Y. Oualil, and D. Klakow, "Approximated and domain-adapted LSTM language models for first-pass decoding in speech recognition," *Proc. of INTERSPEECH 2017*, pp. 2720–2724, 2017.

[20] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *1995 International Conference on Acoustics, Speech, and Signal Processing, ICASSP '95, Detroit, Michigan, USA, May 08-12, 1995*, 1995, pp. 181–184.

[21] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "OpenFst: A general and efficient weighted finite-state transducer library," in *Proc. of the International Conference on Implementation and Application of Automata*. Springer, 2007, pp. 11–23.

[22] M. Kleinert, H. Helmke, G. Siol, H. Ehr, M. Finke, Y. Oualil, and A. Srinivasamurthy, "Machine Learning of Controller Command Prediction Models from Recorded Radar Data and Controller Speech Utterances," in *Proc. of the 7th SESAR Innovation Days (SID)*. University of Belgrade, Nov. 2017.

[23] M. Kleinert, H. Helmke, G. Siol, H. Ehr, A. Cerna, C. Kern, D. Klakow, P. Motlicek, Y. Oualil, M. Singh, and A. Srinivasamurthy, "Semi-supervised Adaptation of Assistant Based Speech Recognition Models for different Approach Areas," in *Proc. of the 37th AIAA/IEEE Digital Avionics Systems Conference (DASC)*, London, UK, 2018.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *Proc. of the IEEE workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE Signal Processing Society, 2011.

[25] Y. Oualil, D. Klakow, G. Szaszák, A. Srinivasamurthy, H. Helmke, and M. P., "A Context-aware Speech Recognition and Understanding System for Air Traffic Control Domain," in *IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, Okinawa, Japan, 2017.