INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

webASR 2 - Improved cloud based speech technology

Thomas Hain, Jeremy Christian, Oscar Saz, Salil Deena, Madina Hasan, Raymond W. M. Ng, Rosanna Milner, Mortaza Doulaty, Yulan Liu

Speech and Hearing Group, Department of Computer Science, University of Sheffield, UK

{t.hain, jchristian1, o.saztorralba, s.deena, m.hasan, wm.ng, rmmilner2, mortaza.doulaty, yulan.liu}@sheffield.ac.uk

Abstract

This paper presents the most recent developments of the webASR service (www.webasr.org), the world's first web–based fully functioning automatic speech recognition platform for scientific use. Initially released in 2008, the functionalities of webASR have recently been expanded with 3 main goals in mind: Facilitate access through a RESTful architecture, that allows for easy use through either the web interface or an API; allow the use of input metadata when available by the user to improve system performance; and increase the coverage of available systems beyond speech recognition. Several new systems for transcription, diarisation, lightly supervised alignment and translation are currently available through webASR. The results in a series of well–known benchmarks (RT'09, IWSLT'12 and MGB'15 evaluations) show how these webASR systems provides state–of–the–art performances across these tasks.

Index Terms: cloud based speech technology, speech recognition, Speech API

1. Introduction

Systems based on Automatic Speech Recognition (ASR) and, in general, Spoken Language Technologies (SLT) have in recent years achieved a level of technological maturity that allows deployment in mass market applications. Such progress is due to substantial investment by industry, and recent advances in machine learning techniques. The number of users of ASR and SLT applications has multiplied, many users have made the technology part of their everyday life. This has resulted in an increase in the number of research groups working in the field worldwide, as well as in an increase in non–experts being interested in developing SLT–based systems. SLT systems however are more complex than ever, and it takes considerable experience to develop systems that yield solid and reliable performance. For those reasons, it is important that the research community as a whole shares and facilitates the use of highly performing research systems with the rest of the community and the general public. This shared knowledge will help the next generation of speech scientists and research groups to continue developing further advances that will keep advancing the field.

The Speech Recognition Virtual Kitchen[1] [1, 2, 3] aims to be a hub where ASR and SLT systems are shared as Virtual Machines (VMs) or cloud–based engines [4]. Research groups can provide VMs to be downloaded from the website or user credentials for cloud–based systems. On the other end, educational and research users can install the VMs or cloud systems and have access to the best performing systems without the long development process that they usually entail.

[1] http://speechkitchen.org

In 2008, webASR became the world's first cloud–based speech recognition engine [5, 6, 7]. It provided a web interface where users could freely sign up and submit their audio files for transcription with one of the available systems developed at the University of Sheffield. While webASR has attracted well over 500 users, advances in web service technologies, as well as advances in core SLT techniques, had made it necessary to implement newer improvements to the service that make it more flexible and easier to access. This paper details these improvements and argues how the cloud architecture of webASR can be used in conjunction with hubs like the Speech Recognition Virtual Kitchen.

The remainder of the paper is organised as follows: Section 2 reviews the original setup and concepts of the webASR service; with Section 3 providing a description of the newest features implemented in the updated version of webASR. Then, Section 4 gives a summary of the existing systems in webASR for transcription, diarisation, alignment and transcription; for which benchmark results are given in Section 5. Finally, Section 6 provides the final remarks and conclusions to the paper.

2. webASR - Version 1

The webASR service web frontend, as originally deployed in 2008, used a standard design method known as the model–view–controller pattern and was implemented entirely in Java. A Servlet, a special type of Java class (conforming to the Java Servlet API) which allows it to respond to HTTP requests, acted as the controller, providing a centralised point of control for all page requests. Since it was necessary to perform a number of client–side analyses on the files chosen for upload by the user (audio type, file size, etc.) it was decided to implement a Java Applet that could manage file checking and restart of uploads. The use of the applet also allowed the upload process to be obfuscated to reduce chances of malicious use.

The primary mode of interaction with the service was via a browser–based interface. This interface implemented both user and administrator functionalities. Upon registration, a regular user could upload files via the Java Applet and retrieve the transcriptions of those files. Administrators, a special type of users, had extended functionalities that allowed them to deploy new systems, give users access to specific systems and manage processed files. An Application Programming Interface (API) was later added as an alternative to the web interface, for integration of the service into applications [8]. In the API, users could authenticate into the system and retrieve a session token that would allow them to submit files for transcription. The same session token could be used to poll the status of the processing and, once the file had finished being recognised, retrieve the transcription output. However, due to its later implementation the API was not intertwined with the user database and specific

http://dx.doi.org/10.21437/Interspeech.2016-700

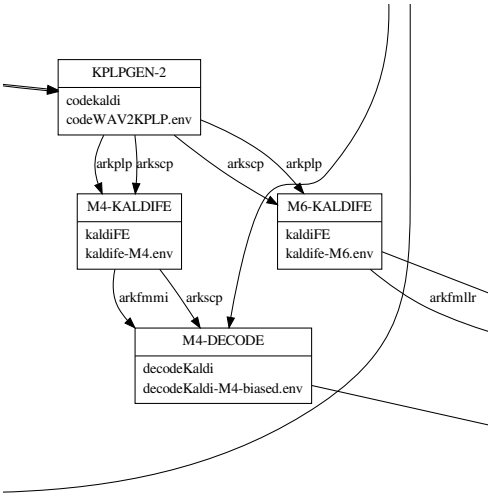

Figure 1: Detail view of an ROTK system diagram

arrangements had to be made to allow users API access, resulting in difficult and inconsistent use of the API. Most users were thus not given access to that functionality.

2.1. System implementation for diverse purposes

One of the key developments allowing for webASR to work was the implementation of a highly flexible and scalable speech processing back–end that is hosted by the University of Sheffield. This back–end uses the Resource Optimisation Toolkit (ROTK), a workflow engine developed by the speech team at the University of Sheffield [5, 9] originally for the purpose of efficient and repeatable implementation of evaluation systems. ROTK allows the formulation of functional modules that can be executed in asynchronous fashion using computing grid infrastructure. Systems are defined as a set of modules linked together by directed connections transferring data of specific types. The most complex research systems can have 70 or more module instances and up to 150 connections; such systems can be visualised in a graph such as that shown in Figure 1, which gives a detailed view of 4 modules and their connections in an actual system. It uses metadata to organise how data is processed in an efficient parallelised way simply implemented by defining the graph. Each module can split its own task into several subtasks, based on data, which then can be processed in parallel. The overall dependency structure of these sub–tasks is automatically inferred in the system. ROTK can integrate with any processing regime. The webASR implementation submits jobs on a grid system using the Open Grid Scheduler engine[2]. ROTK allows for simple repeatability of the experiments as the same graph can be executed on multiple datasets such as development and evaluation sets.

3. Extending the scope - Version 2

The new implementation of the webASR service aimed to avoid the limitations imposed by the initial webASR development, as well as trying to provide further functionalities for the end users.

In order to facilitate its use, webASR has been re–developed following a Representational State Transfer (REST) architecture [10] to handle client/server communication through HTTP requests. This implementation is using the Django web framework[3], which results in simpler and faster deployment than the previous Java Servlet framework. REST architectures are currently used in many web services and are useful for creating programmatic access to applications across a widely recognised protocol. In REST services, an API is simply a specification of remote calls exposed to the customer, which facilitates the experience of the end user. The use of REST principals allows clients to communicate with webASR without prior knowledge of the API structure. Instead the server provides the information the client requires to use the API through regular HTML forms, which can also be used without a web browser application. During the development stage, REST came to be the preferred option as HTTP requests can easily be made from many environments and programming languages that include their own web client packages for such requests. Audio files and metadata are received in POST data and transferred locally to the back–end across an SSH connection. The back–end responsible for processing uses the same ROTK framework as used in the original webASR [5]. Once the recognition process is finished the transcript is stored on webASR and can be retrieved through a GET request. The API handles both user validation and the selection of the system to be used on the back–end.

The second improvement developed in the new webASR implementation involves the use of metadata provided by the user. In many cases users might be able to provide metadata about the audio files they want to process. This metadata can be a manual speech segmentation and/or information about the speakers; a rough transcript or a summary of the content; or certain knowledge about the recording conditions (microphone type, background noise, etc). Previously, webASR did not have the opportunity to use this metadata, but the new implementation allows for the user to upload an XML file with such metadata and the requested system will make use of those parts of such metadata which are considered useful. Currently, two types of metadata are exploited by webASR: First, manual speech segmentation can be provided by the user and be used instead of the automatic segmentation built–in; second, a rough transcript or summary can be provided by the user and language model adaptation will be performed using such data.

The final change for the new webASR is its ability to perform more tasks than simply speech recognition as it was originally designed for. Using the modular ROTK implementation described previously it is possible to build any type of systems beyond ASR; moreover, the output of such systems, whether it is text in any language or speaker or environment information, can be encoded in an XML schema that can be stored by webASR and given back to the user. With this, the new webASR currently provides 4 distinct tasks: Speech transcription, its original task; speech diarisation, for discerning who spoken when; lightly supervised alignment, to align text to a large audio file; and spoken language translation, to provide the translation to a foreign language of an audio file. However, given the modularity of ROTK for implementing any type of system more tasks can be expected to be delivered in the future.

4. Implemented systems in webASR

Currently, webASR provides a variety of systems organised around 3 domains and 4 tasks. As described in Sections 2 and 3, the use of the latest webASR infrastructure together with the ROTK back–end setup allows for a quick development of new domains and tasks. The domains currently covered are

[2] http://gridscheduler.sourceforge.net/

[3] https://www.djangoproject.com/

meetings, general media and lectures; while the tasks are automatic transcription, segmentation and diarisation, alignment and translation. Different domains provide different tasks depending on the specific requirements of the domain.

4.1. Transcription systems

Making complex ASR systems available was originally the intent of webASR, and, as such, ASR remains the main task in 3 newly developed systems covering 3 domains. All of them present a state–of–the–art speech transcription system, based on the latest research carried out at the University of Sheffield in topics such as Deep Neural Network (DNN) acoustic modelling [11, 12, 13, 14], distant microphone recognition [15], adaptation to noisy environments [16, 17, 18], domain adaptation [19, 20], Recurrent Neural Network (RNN) language modelling [21], N–best re–ranking [22, 23] and sentence–end detection [24, 25].

A new webASR meeting transcription system is built for the recognition of meeting recordings in head–mounted microphones and it performs recognition in both single channel and multi–channel fashion. For single channel files, automatic speech segmentation is performed using an unsupervised iterative speech/non–speech detection system [26] based on the Bayesian Information Criterion (BIC). When multiple channel files are uploaded, more advanced speech segmentation is used that incorporates cross–talk information to more accurately detect speech from the target speaker in each channel [27]. 3–pass speech recognition is then performed on found segments of speech. In the first pass, a unified decoding result is generated using a speaker independent system using Perceptual Linear Prediction (PLP) features with Cepstral Mean and Variance Normalisation (CMVN) and a global Heteroscedastic Linear Discriminant Analysis (HLDA) transformation over Minimum Phone Error (MPE)–trained Gaussian Mixture Model (GMM)–Hidden Markov Model (HMM) crossword triphone models. These hypothesis transcripts are used for inferring the Vocal Tract Length Normalisation (VTLN) warp factors of each speaker. The second decoding pass uses VTLN and CMVN normalised filterbank features on MPE–trained DNN–GMM–HMM models. The improved hypotheses are used to estimate cascading speaker–based Maximum Likelihood Linear Regression (MLLR) and Constrained MLLR (CMLLR) transformations, which are then used improve the final decoding using the same features and models from the second pass. All decodings are performed using language models based on Weighted Finite State Transducers (WFST). A further description of this system was published in [15].

A recent focus at Sheffield University has been the development of systems for broadcast media. The media transcription system performs automatic speech segmentation based on the output posteriors of a DNN trained to discriminate speech from non–speech frames. Using these speech segments, the first stage of decoding uses a speaker independent DNN–GMM–HMM model. Its output is used to resegment the audio and cluster speakers using the BIC criterion. Based on the first hypothesis and the speaker segmentation, three speaker dependent recognition systems are run. The first one is a feature Maximum Mutual Information (fMMI)–normalised DNN–HMM system, the second one is a feature–space MLLR (fMLLR) normalised DNN–HMM system, and the third one is a DNN–GMM–HMM with cascading CMLLR and MLLR transformations. The hypotheses of these 3 systems are combined in a Recognition Output Voting Error Reduction (ROVER) framework. All decodings are peformed using 4–gram language models, later re-scored using RNN language models. Full details of this system can be found in the system description for the University of Sheffield system for the MGB challenge [28].

The third new system for lecture transcription system also performs a DNN–based segmentation of speech and non–speech, followed by BIC clustering. This system performs a 2–stage decoding, where in the first stage lattices are created using a Speaker Adaptive Training (SAT) GMM–HMM model in order to compute speaker dependent fMLLR transformations. The second and final stage uses fMLLR–normalised features in a DNN–HMM model. The language model used is a pruned 3–gram, later rescored with a full 4–gram. More details are found on the University of Sheffield submission to IWSLT 2014 [29].

4.2. Segmentation and diarisation systems

Speech segmentation and speaker diarisation is another classical task in SLT. Its goal is to indicate "who speaks when" and, as seen in the descriptions of the previous subsection, it is an integral part of any transcription system. Given the relevance of this task, segmentation and diarisation systems have currently been included as standalone systems in webASR for two different domains: Meetings and general media.

The meeting segmentation system is based on a DNN previously trained on labelled data to distinguish speech and non–speech followed by a HMM to decode the sequence of speech and silence in the audio file.

The diarisation system for general media consists of several stages. First, speech segmentation is performed using the posteriors from a DNN trained to classify speech frames from non–speech frames. This initial segmentation is used to perform speaker independent speech recognition with DNN–HMM models and the hypothesis provided by this system is used to resegment the audio. This latest segmentation is used to do semi–supervised fine–tuning of the DNN model used initially for segmentation, and the fine–tuned DNN is then used to perform another round of improved segmentation. Using this final segmentation, an initial speaker clustering is performed using an unsupervised iterative speaker clustering based on BIC. This speaker segmentation is then used to fine–tune a DNN previously trained to classify a large of training speakers. This fine–tuned DNN is then finally used to extract posterior probabilities for each of the initially found speaker clusters and perform re–clustering. More details are in the University of Sheffield submission for the MGB challenge [30].

4.3. Alignment systems

Lightly supervised alignment is a task where a transcript has to be aligned to an input audio. In cases where the transcript is errorful and unreliable and the audio is acoustically challenging, standard Viterbi alignment techniques fail and a more complex setup is required. Alignment tasks are becoming relevant in certain domains, especially multimedia where it is necessary to deal with unreliable subtitles. This increased importance justified the inclusion of a full lightly supervised alignment system for general media in webASR.

The lightly supervised alignment system implemented makes use of two inputs, the audio file and the transcript to be aligned, which is fed to the system as metadata thanks to the new feature for use of metadata in webASR. The system performs what is called as lightly supervised decoding using a speech recognition system with the same architecture as the transcription system for general media, but with 4–gram language models adapted by interpolating the background models with the transcript for alignment. The output of the lightly supervised decoding is recursively aligned to the input transcript. Finally, a post–processing stage is done based on confidence

measures and regression tree classifiers to identify words in the aligned transcript that do not appear in the spoken audio [31]. This refined aligned transcripts with word–level time boundaries is then provided as the system output.

4.4. Machine translation systems

Machine translation of spoken audio is also another task of growing attention in the community. The ability to recognise and translate speech to a different language is useful for many domains and consistent efforts are being made to develop systems for this task [32]. For that purpose, webASR also includes a machine translation system in the lecture domain.

The translation system performs automatic speech segmentation and recognition using the same setup as the transcription system for lectures presented previously. The output of such system is then cased appropriately using the Moses toolkit [33] and automatic translation to French is done also using Moses with previously trained translation models. The resulting French translation is then provided as output to the user.

5. Benchmark results

All the developed systems in webASR have been tested against well–known benchmarks for each domain and task as part of the development of the systems. This benchmarking not only allows to develop the best performing systems, but also gives users a measure of the quality of each individual system.

The results for the 3 existing transcription systems are presented in Table 1. The meeting transcription system was tested against the NIST Rich Transcription 2009 (RT'09) evaluation set, providing a Word Error Rate (WER) of 28.5%. The lecture transcription system was evaluated on the evaluation data for the ASR track of the IWSLT'12 campaign [34], giving a WER of 12.9%. Finally, the general media transcription system achieved 28.0% WER on the evaluation data for Task 1 of the Multi–Genre Broadcast (MGB'15) challenge [35].

Table 1: *Benchmark results for webASR transcription systems.*

System	Benchmark	Subst.	Delet.	Insert.	WER
Meeting	RT'09	18.4%	6.8%	3.3%	**28.5%**
Lectures	IWSLT'12	8.0%	2.3%	2.6%	**12.9%**
Media	MGB'15	14.1%	10.7%	3.2%	**28.0%**

The two segmentation and diarisation systems, for meetings and general media, were tested against the evaluation sets of the NIST Rich Transcription 2007 (RT'07)[36] and MGB [35] campaigns respectively. The results of these evaluations are shown in Table 2. The meeting segmentation system achieved a 22.5% Segmentation Error Rate (SER) and the general media diarisation system achieved a 49.3% Diarisation Error Rate (DER).

Table 2: *Benchmark results for webASR diarisation systems.*

System	Benchmark	Miss	False	Speaker	SER/DER
Meeting	RT'07	11.8%	10.7%	-	**22.5%**
Media	MGB'15	1.9%	6.4%	41.1%	**49.3%**

The lightly supervised alignment for general media was evaluated on the evaluation data for Task 2 of the MGB challenge [35]. The results in Table 3 show a final F–measure, the metric used in the challenge, of 0.8753.

Table 3: *Benchmark results for webASR lightly supervised alignment system.*

System	Benchmark	Precision	Recall	F–measure
Media	MGB'15	0.8818	0.8689	**0.8753**

Finally, the machine translation system for lectures was evaluated on the English to French translation task of the IWSLT'12 evaluation campaign [34]. The system was scored in the true–cased no–punctuated condition, obtaining a BLEU score of 31.28 as shown in Table 4. Due to the scoring requirements of the machine translation output, human segmentation was used to perform ASR of this dataset, thanks to the metadata feature of webASR, with a WER of 12.5%.

Table 4: *Benchmark results for webASR machine translation system (English to French).*

System	Benchmark	WER(English)	BLEU(French)
Lectures	IWSLT'12	12.5%	**31.28**

5.1. Use of input metadata

One of the main improvements in the new webASR implementation is the option for the user to include input metadata with the file submitted. The current metadata allowed are speech segmentation and a transcription for language model adaptation. Table 5 shows the results of the media transcription system when both types of metadata are used. The WER achieved of 24.1% is 4% absolute better than the one obtained without metadata in Table 1, which shows how the use of metadata can significantly improve the performance that the users can get when they have such metadata available.

Table 5: *Benchmark results for webASR transcription systems using input metadata.*

System	Benchmark	Subst.	Delet.	Insert.	WER
Media	MGB'15	11.9%	10.3%	1.9%	**24.1%**

6. Conclusions

This paper has described the latest improvements in webASR, which was the world's first cloud–based speech recognition engine. The first of these improvements has been the transition to a new web development that uses a REST architecture. This allows to have an easier experience both in the web interface and the programmatic API. The second of these improvement refer to the development of new strong systems for multiple tasks which are now available through webASR. More systems and tasks will be made available in the future such as a language identification [37]. Furthermore, other languages besides English will be supported.

Finally, the new web implementation of webASR allows for an easy use by users of other platforms for sharing speech technology resources, including the Speech Recognition Virtual Kitchen. Registration to webASR is completely free, and submitting audio files and retrieving results can be easily done by the web interface or programmatically using 3 simple HTTP POST and GET commands. It is expected that the general public and researchers from many domains will be drawn to work with webASR.

7. Data access management

Data related to the RT evaluations can be found in the NIST webpages on `http://www.nist.gov/itl/iad/mig/rt.cfm`. Data related to the MGB challenge is available via special license with the BBC on `http://www.mgb-challenge.org/`. Data related to the IWSLT 2012 evaluation can be found in the workshop pages on `http://hltc.cs.ust.hk/iwslt/`. All system outputs and scoring results are available with DOI 10.15131/shef.data.3437441

8. Acknowledgements

This work was partly supported by the EPSRC Programme Grant EP/I031022/1 (Natural Speech Technology) and the EU FP7 DocuMeet Project.

9. References

[1] F. Metze and E. Fosler-Lussier, "The Speech Recognition Virtual Kitchen: An Initial Prototype," in *Proc. of the 13th Interspeech*, Portland, OR, 2012, pp. 1872–1873.

[2] F. Metze, E. Fosler-Lussier, and R. Bates, "The Speech Recognition Virtual Kitchen," in *Proc. of the 14th Interspeech*, Lyon, France, 2013, pp. 1858–1860.

[3] A. Plummer, E. Riebling, A. Kumar, F. Metze, E. Fosler-Lussier, and R. Bates, "The Speech Recognition Virtual Kitchen: Launch Party," in *Proc. of the 15th Interspeech*, Singapore, 2014, pp. 2140–2141.

[4] O. Klejch, O. Platek, L. Zilka, and F. Jurcicek, "Cloud–ASR: Platform & Service," in *Proc. of the 2014 SLT*, Lake Tahoe, CA, 2014.

[5] T. Hain, A. El-Hannani, S. Wrigley, and V. Wan, "Automatic speech recognition for scientific purposes - webASR," in *Proc. of the 9th Interspeech*, Brisbane, Australia, 2008, pp. 504–507.

[6] S. Wrigley and T. Hain, "Making an automatic speech recognition service freely available on the web," in *Proc. of the 12th Interspeech*, Florence, Italy, 2011, pp. 3325–3326.

[7] ——, "Web-Based Automatic Speech Recognition Service - webASR ," in *Proc. of the 12th Interspeech*, Florence, Italy, 2011, pp. 3265–3268.

[8] R. C. Tucker, D. Fry, V. Wan, S. N. Wrigley, and T. Hain, "Extending Audio Notetaker to Browse WebASR Transcriptions," in *Proc. of the 12th Interspeech*, 2011, pp. 3329–3330.

[9] T. Hain, L. Burget, J. Dines, P. Garner, F. Grezl, A. Hannani, M. Huijbregts, M. Karafiat, M. Lincoln, and V. Wan, "Transcribing meetings with the AMIDA systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 486–498, 2012.

[10] R. Fielding and R. Taylor, "Principled design of the modern Web architecture," in *Proc. of the 22th Int. Conf. on Software Engineering*, Limerick, Ireland, 2000, pp. 407–416.

[11] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *Proc. of the 2014 ICASSP*, Florence, Italy, 2014, pp. 5542–5546.

[12] Y. Liu, P. Karanasou, and T. Hain, "An Investigation Into Speaker Informed DNN Front-end for LVCSR," in *Proc. of the 2015 ICASSP*, Brisbane, Australia, 2015, pp. 4300–4304.

[13] P. Zhang, Y. Liu, and T. Hain, "Semi–Supervised DNN Training in Meeting Recognition," in *Proc. of the 2014 SLT*, Lake Tahoe, CA, 2014, pp. 141–146.

[14] M. Doulaty, O. Saz, R. Ng, and T. Hain, "Latent Dirichlet Allocation based organisation of broadcast media archives for deep neural network adaptation," in *Proc. of the 2015 ASRU*, Scottsdale, AZ, 2015, pp. 130–136.

[15] R. Doddipatla, M. Hasan, and T. Hain, "Speaker Dependent Bottleneck Layer Training for Speaker Adaptation in Automatic Speech Recognition," in *Proc. of the 15th Interspeech*, Singapore, 2014, pp. 2199–2203.

[16] O. Saz and T. Hain, "Asynchronous Factorisation of Speaker and Background with Feature Transforms in Speech Recognition," in *Proc. of the 14th Interspeech*, Lyon, France, 2013, pp. 1238–1242.

[17] ——, "Using Contextual Information in Joint Factor Eigenspace MLLR for Speech Recognition in Diverse Scenarios," in *Proc. of the 2014 Int. Conf. on Acoustic, Speech and Signal Proc. (ICASSP)*, Florence, Italy, 2014, pp. 6314–6318.

[18] O. Saz, M. Doulaty, and T. Hain, "Background–Tracking Acoustic Features for Genre Identification of Broadcast Shows," in *Proc. of the 2014 SLT*, Lake Tahoe, CA, 2014, pp. 118–123.

[19] M. Doulaty, O. Saz, and T. Hain, "Data-selective Transfer Learning for Multi-Domain Speech Recognition," in *Proc. of the 16th Interspeech*, Dresden, Germany, 2016, pp. 2897–2901.

[20] ——, "Unsupervised Domain Discovery Using Latent Dirichlet Allocation for Acoustic Modelling in Speech Recognition n," in *Proc. of the 16th Interspeech*, Dresden, Germany, 2016, pp. 3640–3644.

[21] S. Deena, M. Hasan, M. Doulaty, O. Saz, and T. Hain, "Combining Feature and Model-Based Adaptation of RNNLMs for Multi-Genre Broadcast Speech Recognition," in *Proc. of the 17th Interspeech*, San Francisco, CA, 2016.

[22] R. Ng, K. Shah, W. Aziz, L. Specia, and T. Hain, "Quality estimation for ASR k-best list rescoring in spoken language translation," in *Proc. of the 2015 Int. Conf. on Acoustic, Speech and Signal Proc. (ICASSP)*, Brisbane, Australia, 2015, pp. 5226–5230.

[23] R. Ng, K. Shah, L. Specia, and T. Hain, "Groupwise learning for ASR k-best list reranking in spoken language translation," in *Proc. of the 2016 Int. Conf. on Acoustic, Speech and Signal Proc. (ICASSP)*, Shanghai, China, 2016.

[24] M. Hasan, R. Doddipatla, and T. Hain, "Multi-pass sentence-end detection of lecture speech," in *Proc. of the 15th Interspeech*, Singapore, 2014, pp. 2902–2906.

[25] ——, "Noise-matched training of CRF based sentence end detection models," in *Proc. of the 16th Interspeech*, Dresden, Germany, 2015, pp. 349–353.

[26] M. Huijbregts and F. de Jong, "Robust speech/non-speech classification in heterogeneous multimedia content," *Speech Communication*, vol. 53, no. 2, pp. 143–153, 2011.

[27] J. Dines, J. Vepa, and T. Hain, "The Segmentation of Multichannel Meeting Recordings for Automatic Speech Recognition," in *Proc. of the 7th Interspeech*, Pittsburgh, PA, 2006, pp. 1213–1216.

[28] O. Saz, M. Doulaty, S. Deena, R. Milner, R. W. M. Ng, M. Hasan, Y. Liu, and T. Hain, "The 2015 Sheffield System for Transcription of Multi–Genre Broadcast Media," in *Proc. of the 2015 ASRU*, Scottsdale, AZ, 2015, pp. 624–631.

[29] R. Ng, M. Doulaty, R. Doddipatla, W. Aziz, K. Shah, O. Saz, M. Hasan, G. AlHarbi, L. Specia, and T. Hain, "The USFD SLT system for IWSLT 2014," in *Proc. of the 2014 International Workshop on Spoken Language Translation*, Lake Tahoe, CA, 2014.

[30] R. Milner, O. Saz, S. Deena, M. Doulaty, R. W. M. Ng, and T. Hain, "The 2015 sheffield system for longitudinal diarisation of broadcast media," in *Proc. of the 2015 ASRU*, Scottsdale, AZ, 2015, pp. 632–638.

[31] J. Olcoz, O. Saz, and T. Hain, "Error correction in lightly supervised alignment of broadcast subtitles," in *Proc. of the 17th Interspeech*, San Francisco, CA, 2016.

[32] S. Matsuda, X. Hu, Y. Shiga, H. Kashioka, C. Hori, K. Yasuda, H. Okuma, M. Uchiyama, E. Sumita, H. Kawai, and S. Nakamura, "Multilingual Speech-to-Speech Translation System: VoiceTra," in *Proc. of the IEEE 14th International Conference on Mobile Data Management (MDM)*, Milan, Italy, 2013, pp. 229–233.

[33] P. Koehn, H. Hoang, A. birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: open source toolkit for statistical machine translation," in *Proc. of the 45th A. Meeting of the ACL*, Prague, Czech Republic, 2007, pp. 177–180.

[34] M. Federico, M. Cettolo, L. Bentivogli, M. Paul, and S. Stuker, "Overview of the IWSLT 2012 Evaluation Campaign," in *Proc. of the 2012 International Workshop on Spoken Language Translation (IWSLT)*, Hong Kong, 2012.

[35] P. Bell, M. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, and M. Wester, "The MGB Challenge: Evaluating multi-genre broadcast media recognition," in *Proc. of the 2015 ASRU*, Scottsdale, AZ, 2015, pp. 687–694.

[36] J. Fiscus, J. Ajot, and J. Garofalo, "The Rich Transcription 2007 Meeting Recognition Evaluation," *Lecture Notes in Computer Science*, vol. 4625, pp. 373–389, 2008.

[37] R. Ng, M. Nicolao, O. Saz, M. Hasan, B. Chettri, M. Doulaty, T. Lee, , and T. Hain, "Sheffield LRE 2015 System Description," in *Proc. of the Odyssey: The Speaker and Language Recognition Workshop*, Bilbao, Spain, 2016.