Conversational Agent for Daily Living Assessment Coaching Demo

Aditya Gaydhani*
Dept. of Computer Science and Engineering
University of Minnesota
gaydh001@umn.edu

Raymond Finzel*
Dept. of Pharmaceutical Care & Health Systems
University of Minnesota
finze006@umn.edu

Sheena Dufresne
Dept. of Experimental and Clinical Pharmacology
University of Minnesota
gahmx008@umn.edu

Maria Gini
Dept. of Computer Science and Engineering
University of Minnesota
gini@umn.edu

Serguei VS Pakhomov
Dept. of Pharmaceutical Care & Health Systems
University of Minnesota
pakh0002@umn.edu

Abstract

Conversational Agent for Daily Living Assessment Coaching (CADLAC) is a multi-modal conversational agent system designed to impersonate "individuals" with various levels of ability in activities of daily living (ADLs: e.g., dressing, bathing, mobility, etc.) for use in training professional assessors how to conduct interviews to determine one's level of functioning. The system is implemented on the MindMeld platform for conversational AI and features a Bidirectional Long Short-Term Memory topic tracker that allows the agent to navigate conversations spanning 18 different ADL domains, a dialogue manager that interfaces with a database of over 10,000 historical ADL assessments, a rule-based Natural Language Generation (NLG) module, and a pre-trained open-domain conversational sub-agent (based on GPT-2) for handling conversation turns outside of the 18 ADL domains. CADLAC is delivered via state-of-the-art web frameworks to handle multiple conversations and users simultaneously and is enabled with voice interface. The paper includes a description of the system design and evaluation of individual components followed by a brief discussion of current limitations and next steps.

1 Introduction

A person's ability to function independently in everyday life depends on multiple factors including, but not limited to, intact physical and mental capacity. In the United States, significant public resources are dedicated to providing assistance to those in need. A key aspect of assistance programs is to provide ongoing assessment of individuals to determine their level of functioning (e.g. independent, needs supervision, needs physical assistance, or dependent) and their specific needs in order to provide assistance appropriately. These assessments are conducted by certified assessors specifically trained for this purpose. A challenge in the assessment process is to ensure consistency across large numbers of assessors with various degrees of experience and interview skills and to prepare novice assessors for the variety of interactions they will experience in the field. The Conversational Agent for Daily Living Assessment Coaching (CADLAC) is designed to coach certified assessors to conduct their assessment interviews in a natural conversational style that simulates real interactions. Previously, dialogue systems similar to CADLAC have been developed (Campillos Llanos et al., 2015; Nirenburg et al., 2008; Jaffe et al., 2015; Laleye et al., 2020). These systems simulate "Virtual Patients", which are used in healthcare education. CADLAC is tailored to support novel application domains of function and disability. An example of the interaction with the conversational agent is shown in Figure 1. The interface and a video highlighting the system can be found here [1] [2].

2 Data

We used two sources of data in order to inform CADLAC system design, train machine learning models, and to develop a database to support rule-based approaches used by the system. One source of data consisted of a survey that was administered to certified assessors, and the other consisted

*Equal contribution.

[1] Demo: https://rxinformatics.net/cadlac
[2] Video: https://vimeo.com/500734362

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, pages 321–328
April 19 - 23, 2021.

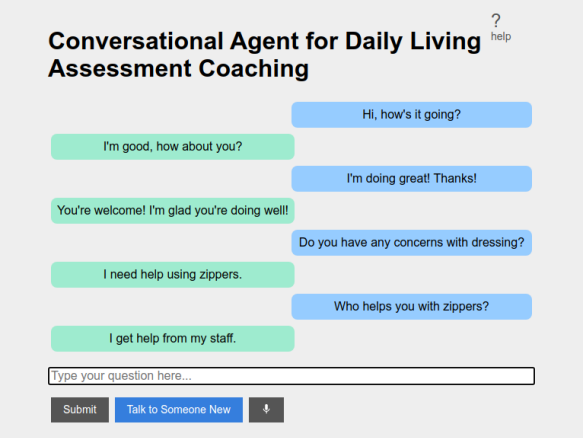

Figure 1: Example of interaction with the conversational agent.

of anonymized historical assessment data shared by the Minnesota Department of Human Services (DHS).

2.1 Survey Data

We designed a survey to collect sample dialogues from certified assessors. This survey was administered to approximately 1,700 assessors statewide. The assessors were asked to recall some of their past assessments and provide examples of interactions that they had with people during the assessment interviews. Specifically, each example consists of up to 3 dialogue turns between the assessor and the person being interviewed, the gender and age category of the person, domain of the conversation, and the person's ability level within the domain. The data consists of assessments of activities of daily living (ADLs - e.g., walking) and instrumental activities of daily living (iADLS - e.g., paying bills) in 18 functional domains related to personal cares, movement, household management, and eating/meal preparation. We also manually annotated the assessor questions for 6 intents: challenges, preferences, equipment, helper, generic, and frequency. We were able to collect a total of 2,885 dialogues through the survey. A sample record from the resulting dataset, including the annotations for intents, is shown in Table 1.

Domain:	Grooming
Ability:	Physical Assistance
Assessor-1:	"Can you tell me about how you take care of your grooming needs?" *intent - generic*
Participant-1:	"I have a hard time."
Assessor-2:	"Can you brush your hair?" *intent - challenges*
Participant-2:	"No, I can't reach my hair to get it brushed in the back."
Assessor-3:	"Who helps you to brush your hair?" *intent - helper*
Participant-3:	"My daughter helps me to brush my hair."
Age:	65-84
Gender:	Female

Table 1: Example dialogue from the survey.

2.2 Synthetic Profiles

CADLAC relies on a database of over 10,000 historical assessments, conducted by experienced certified assessors and managed by Minnesota DHS. Each historical assessment contains fields that indicate the person's ability to function in ADLs and iADLs in addition to basic demographic information such as age range and sex of the person being assessed. It also contains certified assessors' notes taken during the assessments. These notes represent very brief descriptions of the assessed person's challenges, preferences, and equipment they use to help them, among other information organized by the ADL and iADL domain.

Historical data was anonymized by DHS staff for inclusion in CADLAC by removing any individually identifiable information including individuals' names and exact age information that was converted to age ranges. Furthermore, sensitive personal information such as phone number, email, location, etc. was excluded from the historical data, keeping the privacy of the individuals protected.

These anonymized historical assessments are used to generate synthetic profiles of "individuals" that specify varying levels of independence in everyday functioning and specific needs. These profiles are created by mapping the categorical attributes related to the independence levels in the historical assessment to those levels specified for the conversational agent (CA). Additionally, assessor notes about challenges, preferences, and equipment from the historical data were populated in the synthetic profiles.

The profiles are used to customize the CA and generate natural language responses that are tailored to the question asked by the assessor and are as consistent as possible with all of the information in the profile. For example, if the synthetic

profile states that the individual being assessed is completely dependent on external assistance in the mobility domain, the responses generated by CADLAC to a question about the ability to perform heavy housekeeping should not indicate any degree of independence in this domain either. The profiles include a numeric representation of the independence level of the "person" represented by the profile. These numeric representations are used to compare assessments produced by novice assessors using CADLAC for training to those produced by experienced assessors, and to provide summary feedback about the assessment.

Despite the fact that these profiles are based on data from real individuals assessed in the state of Minnesota, the profiles may potentially convey biases present in the underlying data. In order to minimize potential systematic bias, the historical data used to construct the profiles were randomly sampled from a diverse population of assessed individuals with equal proportions by sex and with the following race distribution: 17.1% African American; 2.4% American Indian; Asian or Pacific Islander 7.7%; Hispanic 2.6%; White 64.4%; Two or more races 1.1%; and Unknown race 4.6%. The current prototype of CADLAC does not use race information; however, this information is available in the underlying data and can be used to adjust the composition of the synthetic profile database as needed for assessor training purposes.

3 System Design

CADLAC is implemented on the MindMeld platform for conversational AI applications (Raghuvanshi et al., 2018) that relies on a commonly used modular dialogue system design consisting broadly of natural language understanding (NLU), natural language generation (NLG) and a dialogue state tracker/manager (DM) components. These components of CADLAC prototype have been developed using a hybrid machine learning and rule-based approaches. The prototype is currently deployed via a web service written in Python with the modern asynchronous web Responder framework. This web service is responsible for accepting requests from a user-facing web client, managing user sessions, and passing conversation objects into the Dialogue Parser. The web-based client supports text-only, voice-only or hybrid modalities. This demonstration will focus on showing the natural dialogue between human users and CADLAC aimed at assessing the level of functioning of the "individual" impersonated by CADLAC and the feedback provided to the users regarding their assessments. The system architecture is shown in Figure 2.

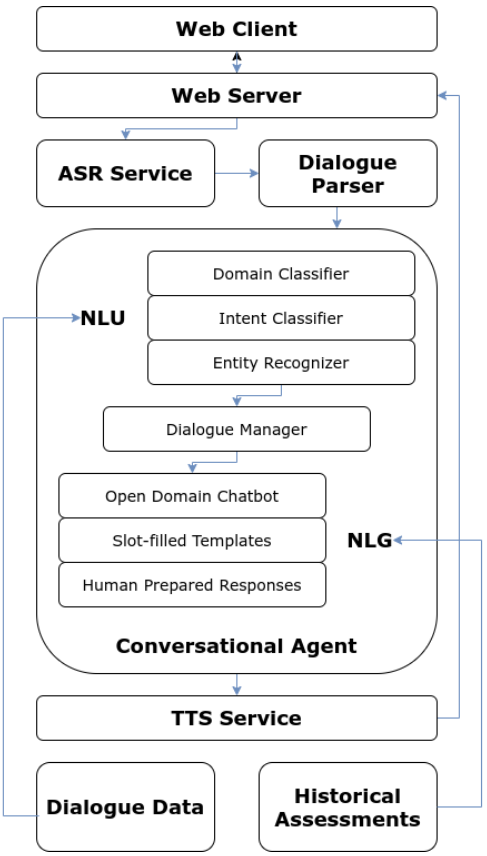

Figure 2: CADLAC system architecture.

4 Natural Language Understanding

4.1 Domain Classifier

The domain classifier (a.k.a. topic tracker) categorizes the input query into one of 18 domains related to ADLs and iADLs, as well as two additional domains: "generic follow-up question" and "unsupported". CADLAC's domain recognizer comprises a BiLSTM neural network (Hochreiter and Schmidhuber, 1997) that we trained on available survey data using GloVe embeddings (Pennington et al., 2014) to represent the semantics of input tokens. We evaluated this model using 10-fold cross-validation resulting in a mean f-score of 0.801 and an accuracy of 0.830 across all domains.

4.2 Intent Classifier

Next, the NLU module recognizes the intent of the user query. In our case, each domain has the following intents that reflect the nature of the ques-

tions asked by assessors: challenges, preferences, generic, equipment, unsupported, helper, and frequency. These intents specify the type of information that the assessor wants to elicit. We used the survey data to train an intent classifier for each domain using the same BiLSTM architecture that we used for the domain recognizer. The results of 10-fold cross-validation for this component consist of a range of f-scores from 0.704 to 0.927 that vary by domain.

4.3 Named Entity Recognizer

We also trained a Named Entity Recognizer to identify the words or phrases, referred to as "entities", present in the input query (e.g., shirt, shoes, pants are entities in the dressing domain). These entities are then used to fill the empty slots in the natural language response or select an appropriate response from the knowledge base. We also use a rule-based language parser within MindMeld to model the dependencies between the recognized entities.

4.4 Dialogue Manager

The dialogue manager consists of the dialogue state tracker, which maps the input query to appropriate dialogue states. Each dialogue state is responsible for handling a particular type of query. We use a rule-based and pattern matching procedure, which depends on the domain and the intent of the input query, to define the dialogue states. One of the important functionalities of the CA is to handle follow-up questions as illustrated in Figure 3. For this purpose, we use the domain of the previous turn and make a transition to the dialogue state specified by the intent of the current turn. If the intent of the question is unsupported, then we use the intent of the previous turn and the domain of the current turn, and make a transition to the corresponding dialogue state. The unsupported queries are handled by the neural model based on GPT-2 (Zhang et al., 2020) as illustrated in Figure 4.

5 Natural Language Generation

We use a rule-based approach in which we first look up a field in the knowledge-base of historical assessments that corresponds to the identified topic and intent for a specific synthetic profile (e.g., challenges[intent] with dressing[domain]). Information contained in historical assessments is underspecified and is not usable as a natural language response. For example, it may contain a note "Behavioral issues" for challenges with dressing. We manually annotated a subset of over 100 assessments, where the annotators were instructed to become familiar with the person's level of functioning in various domains and use that knowledge to convert the historical notes to a format that would sound more natural yet still consistent with the synthetic profile (e.g., "Behavioral issues" note for a 5 year old child's assessment would be converted to "He can't dress by himself because he throws a tantrum each time he has to change clothes.") The current prototype of CADLAC's dialogue manager queries the knowledge base for these manually converted responses and returns a response that most closely matches the named entities mentioned in the user's question. If no natural language response is found, CADLAC generates a generic response randomly chosen from a set of responses consistent with the synthetic profile (e.g., for a profile of a person who requires intermittent physical assistance with dressing, the response may be "I need someone to help me with this"). We are currently experimenting with transformer neural models used in machine translation in order to determine if they can "learn" the mapping between the original historical assessment notes and the natural language responses; however, the current demo does not include these models yet.

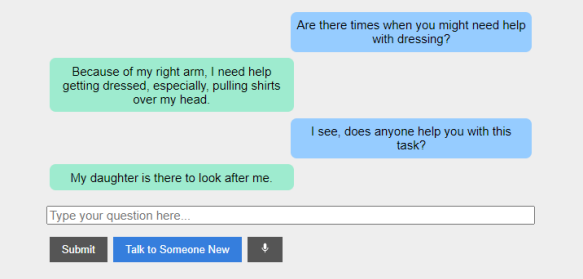

Figure 3: Response to a follow-up question. The second question of the conversation refers to the previous domain of dressing.

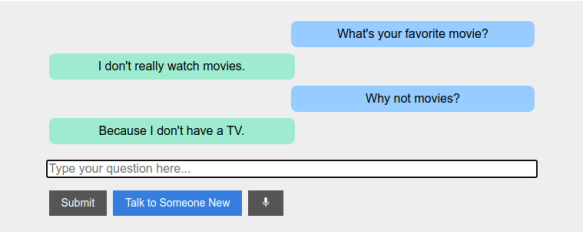

Figure 4: Response to off-topic questions.

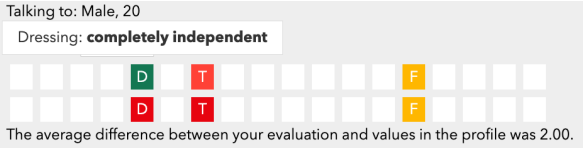

Figure 5: Assessment feedback. Top row shows values in the profile only for those domains assessed up to a checkpoint. Bottom row shows user-selected assessments.

6 Feedback

The feedback to users being trained to perform assessments is provided via a visual interface designed to compare users' assessments to those stored in synthetic profiles as illustrated in Figure 5.

7 Voice Services

In order to enable voice input-output capabilities in CADLAC we implemented a Automatic Speech Recognition (ASR) and a Text-to-Speech (TTS) web services. Both services are implemented using PyTorch.

Voice activity is streamed from the web client to the web server in real time using an implementation of WebRTC peer connections. The WebRTC protocols are available in most modern browsers, and include hooks to access media devices, standards for establishing peer connections, and asynchronous data channels. The implementation of WebRTC that was used for the python web server was AIORTC.

After voice data arrive at the server they are passed to the ASR service, which transcribes English words from the speech utterance. These words take the place of the text from the chat interface for the rest of the conversational turn.

7.1 ASR Service

We trained an ASR system based on Baidu's Deep Speech 2 architecture (Amodei et al., 2016) implemented in PyTorch [3] consisting of 3 convolutional neural network (CNN) layers, followed by 5 bidirectional recurrent neural network (RNN) layers with gated recurrent units (GRU), a single lookeahead convolution layer followed by a fully connected layer and a single softmax layer. The system was trained using the Connectionist Temporal Classification (CTC) loss function (Graves et al., 2006).

In addition to the default greedy search decoding over the hypotheses produced by the softmax layer, the system's implementation also can use a beam search decoder with a standard n-gram language model. We used default hyperparameters: size of the RNN layers was set to 800 GRU units; starting learning rate was set to 0.0003 with the annealing parameter set to 1.1 and momentum of 0.9. Audio signal processing consisted of transforming the audio from the time to the frequency domain via Short-time Fourier transform as implemented by the Python librosa library. The signal was sampled in frames of 20 milliseconds overlapping by 10 milliseconds. The resulting input vectors to the first CNN layer of the Deep Speech 2 network consisted of 160 values representing the power spectrum of each frame.

A collection of speech corpora available from the Linguistic Data Consortium was used as training data. These corpora include the Wall Street Journal (WSJ: LDC93S6A, LDC94S13B), Resource Management (RM - LDC93S3A), TIMIT (LDC93S1), FFMTIMIT (LDC96S32), DCIEM/HCRC (LDC96S38), USC-SFI MALACH corpus (LDC2019S11), Switchboard-1 (LDC97S62), and Fisher (LDC2004S13, LDC2005S13). In addition to these corpora, we used the following publicly available data: TalkBank (CMU, ISL, SBCSAE collections) (MacWhinney and Wagner, 2010), Common Voice (CV: Version 1.0)) corpus [4], Voxforge corpus [5], TED-LIUM corpus (Release 2) (Rousseau et al., 2014), LibriSpeech(Panayotov et al., 2015), Flicker8K(Hodosh et al., 2013), CSTR VCTK corpus (Veaux et al., 2017), and the Spoken Wikipedia Corpus (SWC-English(Köhn et al., 2016)). Audio samples from all of these these data sources were split into pieces shorter than 25 seconds in duration. The total size of the resulting corpus was approximately 4,991 hours of audio (2,000 hours contributed by the Fisher corpus alone). Finally, we also used audio data from various prior studies that were conducted at the University of Minnesota consisting of story recall, verbal fluency, and spontaneous narrative tasks. With the exception of the Fisher and Switchboard corpora, all other data were recorded at a minimum of 16 kHz sampling frequency. The Fisher and Switchboard corpora contain narrow-band telephone conversations sampled

[3] https://github.com/SeanNaren/deepspeech.pytorch

[4] http://voice.mozilla.org

[5] http://www.voxforge.org/

at 8 KHz. All data were either downsampled or upsampled and converted using the SoX toolkit[6] to a single channel 16 bit 16 kHz PCM WAVE format.

The performance of the ASR service was evaluated off-line using the heldout portion of the TED-LIUM corpus. Without using a language model for rescoring the output of the neural model (greedy decoding), the word error rate (WER) and character error rate (CER) of our ASR system were 18.84 and 5.24, which are comparable to those previously reported for the same dataset also using a Deep Speech 2 system (WER: 28.1, CER: 9.2) (Hernandez et al., 2018). Using a 4-gram language model constructed with the SRILM Toolkit (Stolcke, 2002) from the English language portion of the 1 Billion words text corpus[7] model with Kneser-Ney smoothing (Ney et al., 1994) resulted in improving ASR accuracy to WER: 15.73 and CER: 4.57.

7.2 TTS Service

We used a pre-trained model based on Tacotron2 (Shen et al., 2017) and WaveGlow (Prenger et al., 2018) for the text-to-speech service. This model was implemented in PyTorch and is based on the NVIDIA's GitHub repositories for Tacotron2 [8] and WaveGlow [9]. The Tacotron2 model converts the input text to mel spectrograms and then the WaveGlow model uses the mel spectrograms to generate speech. The Tacotron2 implementation used here slightly differs from the one described in by Shen et al. (2017): it uses Dropout (Srivastava et al., 2014) regularization instead of Zoneout (Krueger et al., 2016) for the LSTM layers, and replaces the WaveNet model with the WaveGlow model. The models are trained on the LJ Speech (Ito and Johnson, 2017) dataset using mixed precision training (Micikevicius et al., 2017).

The above model generates speech in female voice since it is trained on the LJ Speech dataset, which has voice samples from a single female speaker. However, our system has synthetic profiles for both males and females. In order to generate speech for a male profile, the current implementation relies on pitch manipulation techniques. Specifically, we use the phonetics software Praat (Boersma and Weenink, 2018) through the library Parselmouth (Jadoul et al., 2018), which exposes the functionality and algorithms of Praat in Python. To change the female voice to a male voice, we set the parameter *formant shift ratio* to 0.85 and *new pitch median* to 100 Hz. The formant shift ratio determines the frequencies of the formants and the new pitch median determines the median pitch of the male voice. Using these specific values of the parameters gives us the best results. However, we are currently exploring ways to retrain the Tacotron2 and WaveGlow model on a male voice dataset to generate better quality outputs.

[6] http://sox.sourceforge.net

[7] https://github.com/ciprian-chelba/1-billion-word-language-modeling-benchmark

[8] https://github.com/NVIDIA/tacotron2

[9] https://github.com/NVIDIA/waveglow

8 Limitations and Future Steps

One of the limitations of the current implementation of CADLAC is that it does not currently learn from user input. One of the next key steps in further development of this system is to implement active learning components for domain and intent classification, ASR, and other supervised components of the system. We are also currently developing a formal evaluation of the usability of this system with human end-users. Specifically, we plan to use metrics of sensibility and specificity for each system response as proposed by Adiwardana et al. (2020) in addition to overall subjective measures of dialogue success, conversation naturalness, and intelligibility of responses. We also plan to evaluate the system for any potential bias in responses generated by the system and develop ways of un-biasing the system via hybrid rule-based and data-driven approaches (Liu et al., 2020).

9 Acknowledgements

The work on this project was supported by funding from the Minnesota Department of Human Services. We would like to thank the people at DSD and MNIT for help with project specifications, gathering of historical data, and expert guidance on domain-specific aspects of the project. We would also like to thank Pamela Miller, Sidney Kiltie, and Elise Moore for help with transforming certified assessor notes to natural language format and Julia Garbuz for helping to develop and conduct the surveys of DHS assessors.

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot.

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, Jie Chen, Jingdong Chen, Zhijie Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Ke Ding, Niandong Du, Erich Elsen, Jesse Engel, Weiwei Fang, Linxi Fan, Christopher Fougner, Liang Gao, Caixia Gong, Awni Hannun, Tony Han, Lappi Vaino Johannes, Bing Jiang, Cai Ju, Billy Jun, Patrick LeGresley, Libby Lin, Junjie Liu, Yang Liu, Weigao Li, Xiangang Li, Dongpeng Ma, Sharan Narang, Andrew Ng, Sherjil Ozair, Yiping Peng, Ryan Prenger, Sheng Qian, Zongfeng Quan, Jonathan Raiman, Vinay Rao, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Kavya Srinet, Anuroop Sriram, Haiyuan Tang, Liliang Tang, Chong Wang, Jidong Wang, Kaifu Wang, Yi Wang, Zhijian Wang, Zhiqian Wang, Shuang Wu, Likai Wei, Bo Xiao, Wen Xie, Yan Xie, Dani Yogatama, Bin Yuan, Jun Zhan, and Zhenyao Zhu. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 173–182. JMLR.org.

Paul Boersma and David Weenink. 2018. Praat: doing phonetics by computer [Computer program]. Version 6.0.37, retrieved 3 February 2018 http://www.praat.org/.

Leonardo Campillos Llanos, Dhouha Bouamor, Éric Bilinski, Anne-Laure Ligozat, Pierre Zweigenbaum, and Sophie Rosset. 2015. Description of the Patient-Genesys dialogue system. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 438–440, Prague, Czech Republic. Association for Computational Linguistics.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 369–376, New York, NY, USA. ACM.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. Ted-lium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In *Speech and Computer*, pages 198–208, Cham. Springer International Publishing.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

M. Hodosh, P. Young, and J. Hockenmaier. 2013. Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Keith Ito and Linda Johnson. 2017. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/.

Yannick Jadoul, Bill Thompson, and Bart de Boer. 2018. Introducing Parselmouth: A Python interface to Praat. *Journal of Phonetics*, 71:1–15.

Evan Jaffe, Michael White, William Schuler, Eric Fosler-Lussier, Alex Rosenfeld, and Douglas Danforth. 2015. Interpreting questions with a log-linear ranking model in a virtual patient dialogue system. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 86–96, Denver, Colorado. Association for Computational Linguistics.

Arne Köhn, Florian Stegen, and Timo Baumann. 2016. Mining the spoken wikipedia for speech data and beyond. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

David Krueger, Tegan Maharaj, János Kramár, Mohammad Pezeshki, Nicolas Ballas, Nan Rosemary Ke, Anirudh Goyal, Yoshua Bengio, Aaron Courville, and Chris Pal. 2016. Zoneout: Regularizing rnns by randomly preserving hidden activations.

Fréjus A. A. Laleye, Gaël de Chalendar, Antonia Blanié, Antoine Brouquet, and Dan Behnamou. 2020. A French medical conversations corpus annotated for a virtual patient dialogue system. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 574–580, Marseille, France. European Language Resources Association.

Haochen Liu, Jamell Dacon, Wenqi Fan, Hui Liu, Zitao Liu, and Jiliang Tang. 2020. Does gender matter? towards fairness in dialogue systems.

Brian MacWhinney and Johannes Wagner. 2010. Transcribing, searching and data sharing: The CLAN software and the TalkBank data repository. *Gesprachsforschung: Online-Zeitschrift Zur Verbalen Interaktion*, 11:154–173.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. 2017. Mixed precision training.

Hermann Ney, Ute Essen, and Reinhard Kneser. 1994. On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language*, 8:1–38.

Sergei Nirenburg, Stephen Beale, Marjorie McShane, Bruce Jarrell, and George Fantry. 2008. Language understanding in Maryland virtual patient. In *Coling 2008: Proceedings of the workshop on Speech Processing for Safety Critical Translation and Pervasive Applications*, pages 36–39, Manchester, UK. Coling 2008 Organizing Committee.

V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. 2018. Waveglow: A flow-based generative network for speech synthesis.

Arushi Raghuvanshi, Lucien Carroll, and Karthik Raghunathan. 2018. Developing production-level conversational interfaces with shallow semantic parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 157–162.

Anthony Rousseau, Paul Deléglise, and Yannick Estève. 2014. Enhancing the ted-lium corpus with selected data for language modeling and more ted talks. In *LREC*.

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. 2017. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Andreas Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *INTERSPEECH*.

Christophe Veaux, Junichi Yamagishi, and Kirsten Macdonald. 2017. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. Dialogpt: Large-scale generative pre-training for conversational response generation. In *ACL, system demonstration*.