Navajo Speech Recognition Using Low-Resource Language Models

Emery M. Sutherland
Dept. of Electrical Engineering
University of New Mexico
Albuquerque, United States
emsland@unm.edu

Melvatha R. Chee
Dept. of Linguistics
University of New Mexico
Albuquerque, United States
mchee@unm.edu

Marios S. Pattichis
Dept.of Electrical Engineering
University of New Mexico
Albuquerque, United States
pattichis@unm.edu

2024 58th Asilomar Conference on Signals, Systems, and Computers | 979-8-3503-5405-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/IEEECONF60004.2024.10942828

***Abstract*— The paper describes the development of a speech recognition system to classify Navajo (Diné) words using Low Resource Language (LRL) datasets. There are presently no known high-quality open-sourced datasets for the Diné language that are needed to train models for speech Recognition. A small dataset was designed to train several models. To overcome the scarcity of the LRL dataset, the audio recordings were augmented to account for time-stretching, amplitude variations, time shifts, and small amounts of white Gaussian noise.**

Several models were trained using different optimization methods. The models included a Recurrent Neural Network (RNN), a Convolutional Neural Network (CNN), and a Long Short-Term Memory (LSTM) model. The results compare nine methods: SGD, Momentum, Nesterov, AdaGrad, RMSProp, Adam, Adamax, Nadam, and AdamW. For the best model, we report a family of different training/validation curves. The results demonstrate excellent classification performance on LRL models for Navajo Speech Recognition.

Keywords—Diné, Navajo Speech Recognition, Low Resource Language Datasets

I. Introduction

Automatic speech recognition systems are commonly deployed for use with many IOT devices. The languages supported in these devices are based on large datasets for which mature automatic speech recognition systems have been developed. For this paper, we consider the development of an automated speech recognition system for Diné Bizaad, which does not have a large, high-quality dataset for training.

The lack of high-quality curated digital resources can be attributed to several factors. One is, it would cost a lot of money to record and digitize Navajo language audio and written materials. It would also take a team of skilled people to get a significant amount of work done for this. Finding people with the rights skill set or training others to learn the skills is costly. Thus, it will take a significant amount of funding.

Additionally, the Navajo language was classified by the U.S. military until 1968 [4]. This was due to the success of the language as an unbreakable code during World War II. Today, the Diné language is classified as endangered with few monolingual speakers. Also, the written form of the language was not created by the Diné people. Robert Young and William Morgan Sr. developed the Navajo orthography using Romanized letters [6]. The Navajo writing system, developed for the Diné people, was forced upon them to use by making it the only available writing system. The written form of the language began to gain popularity sometime after the U.S. military declassified it. Navajo people began to realize that the written form could be a useful tool [4]. Essentially, one could take an entire English sentence and create a single complex words to express the same concept in Diné Bizaad. This was accomplished through multiple morphemes usage.

The Diné people speak a Polysynthetic language with a rich verbal morphological system. Developing an automatic speech recognition system (ASR) is particularly challenging due to the number of inflections in the language. The number of inflections causes a simple word to take on a new meaning and adds to the complexity of translation, as discussed in [7]. The difficulty with training an ASR with a LRL has been discussed in [8].

For this paper, we were interested in developing an ASR system for the Diné language that can also be used for STEM+C education. Thus, we designed a small dataset of 12 vocabulary words that can be used to control a small robot (see Table I). Our work was motivated and funded through the NSF ESTRELLA project that focuses on the development of STEM+C curriculum for bilingual children (eg. See [9-16]).

We investigate the successful training of three deep-learning architectures based on RNN, CNN, and LSTM. Overall, we have found that the LSTM architecture performed the best. We hope that the lessons learned from our study will inform groups working on ASR system for Native American languages with limited resources. Thus, we provide detailed information on dataset collection and LSTM model optimization using various optimization methods.

The rest of the paper is organized into four sections. In section II, we provide more details on our dataset. In section III, we describe the deep learning architectures that were developed to recognize our limited vocabulary. Section IV provides detailed results based on extensive optimization of our proposed deep learning architectures. We provide concluding remarks in section V.

II. Dataset

This section provides more details of how we constructed our dataset.

A. Diné Word Selection

As mentioned earlier, our goal was to construct a limited vocabulary for STEM+C education. Our chosen words in Table I contain common greetings and commands for controlling a rover. In addition, we restricted our attention to words that can be uttered in less than a second. The words were chosen by Dr. Chee, our co-author, and Director of the Navajo Language Program at the University of New Mexico.

Since the Diné language is Polysynthetic, some of the potential rover commands were difficult to say and required more than one second to pronounce. The rover command to turn right in Diné is nish'náájígo which translates to English as "on the right side." Dr. Chee suggested removing the prefix and suffix leaving the root direction, right. It would sound weird in Diné because directions are stated of who or what is to the right of the object, but the direction is still present. As a result, our initial vocabulary of twenty-five potential words was narrowed down to twelve words.

B. Word Collection

Each word was repeated twenty times by ten Native American volunteers. The volunteers for the Diné recordings determined how many times each word would be repeated. Sampling words were recorded from six males and four females. Thus, we created a fully balanced dataset based on twenty utterances of 12 words, repeated by ten speakers.

Early in our experiments, we found that repeating each word twenty times resulted in speaker fatigue. As a result, pronunciation, as well as annunciation, began to fade. Based on earlier work in [18], to reduce speaker fatigue, we decided to record the twenty utterances using multiple recordings of a small number of utterances.

C. Word Collection

Our primary goal was to develop an ASR system that can be applied to new student speakers without retraining. Thus, for training and testing our system we selected words by different speakers. We randomly selected three speakers for testing. Then, we created three training-testing datasets based on testing each one of the speakers while training using the remaining nine speakers.

TABLE I. 12 DINÉ VOCABULARY WORDS

Diné word	*English equivalent*
yá'át'ééh	hello
hágoónee'	farewell
ádin	nothing (zero)
náás	forward
k'adí	enough (stop)
tł'ahjí	left side
náájí	right side
saad	Language/word
t'ááłá'í	one
naaki	two
táá'	three
díí'	four

III. METHODOLOGY

We summarize our proposed approach using three steps. First, we applied data augmentation. Second, we performed feature extraction for each word. Third, we processed the extracted features using three different deep-learning architectures. For training, we considered using nine different optimization methods, as summarized in Table II. In what follows, we provide more details for each step.

For data augmentation, we considered several different transformations. Specifically, we applied time-stretching, amplitude variations, time shifts, small amounts of white Gaussian noise, and their combinations. Using data augmentation, we doubled each training dataset. We did not apply any data augmentation to our testing dataset. Instead, we report results on our raw dataset extracted from a speaker who was not part of our training dataset.

For feature extraction, we first applied a lowpass filter that removed frequencies above 8Khz. Then, we computed 63x63 Mel-Spectrograms from each audio recording.

The extracted features were then classified using deep-learning architectures based on CNN, RNN, and LSTM. As summarized in Table II, for training, we investigated the use of RMSProp, Adam, AdamW, Adamax, Nadam, Nesterov, AdaGrad, Momentum, and SGD.

We provide system diagrams for three deep-learning architectures in Fig. 1. The three final networks of Fig. 1 represent our optimized models based on our extensive experimentation with the size of the Mel-Spectrogram image, the number of layers, the number of neurons used in each layer, dropout rates, and the activation functions.

To improve system performance, changes were made to the following: layers, neurons, batch size, optimizer values, activation, dropout, and mel count. After experimentation, some models did not improve after changes were made. For each network, we used the optimization settings given in Table II and also considered different batch sizes. A batch size of 32 was considered optimal.

TABLE II. OPTIMIZERS

Optimizer	*Values*
RMSProp	lr = 0.001
Adam	lr = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$
AdamW	lr = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$
AdaGrad	lr = 0.01
Adamax	lr = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$
Nadam	lr = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$
Nesterov	lr = 0.01, p = 0.9
Momentum	lr = 0.01, p = 0.9
SGD	lr = 0.01, p = 0.9

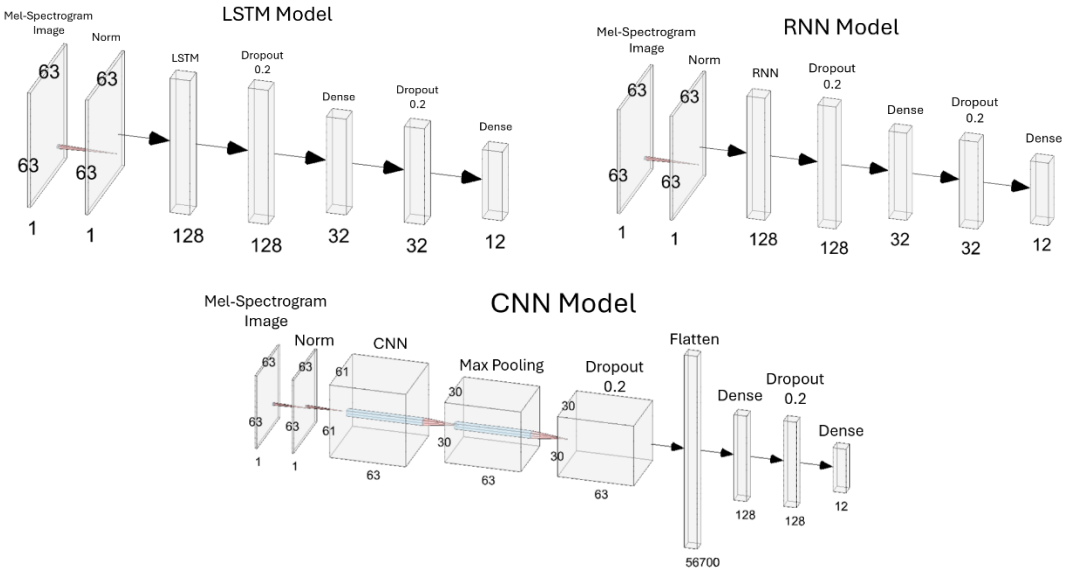

Fig. 1. Training models.

IV. Results

We begin with a summary of our results using the RNN, CNN, and the LSTM networks. In terms of performance, RNN performed the worst. CNN did better than RNN but performed significantly worse than the LSTM network.

We were not able to get the RNN to converge to anything meaningful. We could not get the training and validation losses to converge with additional epochs. The overall accuracy was around 9.0%.

We performed extensive experimentation with the CNN models. Initially, we tested our CNN model with four words. However, we saw a significant drop in performance when we tested the network on twelve words. The validation loss remained above 0.03 for all optimization methods. Furthermore, we could not improve the results by varying our models. Overall, the CNN system gave accuracy scores ranging from 70% to 90%. Training loss approached very low values while validation losses converged around 0.04. In terms of overall performance, the CNN performed reasonably well. Confusion matrices did not show any significant biases in the model. Total training time was about 20 minutes to train using a single optimizer on an Intel i5 8600 processor with 16 GB of RAM.

The LSTM models gave the best results. Our training loss approached low values as shown in Fig. 2. Our validation loss approached very low values as shown in Fig. 3. In terms of optimization methods, AdamW, Adam, Nadam, and RMSProp consistently outperformed all other methods. However, it is important to note that all optimization methods performed relatively well with the LSTM network. It took about 10 minutes to train using a single optimizer.

For the LSTM network, we present confusion matrix results for two of the three speakers. Overall, the network gave more than 90% accuracy for all three speakers in our leave-one-out validation test. As shown in Fig. 4, LSTM gave perfect results for one of the speakers. For this example, LSTM was trained using RMSProp. The worst results are shown in Fig. 5. For this example, the network was trained using Momentum. From the example, it is clear that the error is due to the misclassification of three words. For the rest of the words, the system achieved perfect classification results.

The misclassification of the words yá'át'ééh and saad was due to speaker pronunciation as seen in Fig 5. The speaker frequently did not emphasize the "t" sound in yá'át'ééh and would often end the word with an "i" sound like the ending of naaki. This is why some of the testing set for yá'át'ééh were misclassified. A similar circumstance is seen with saad. The speaker did not emphasize the "d" at the end of the word causing a confusion with the classification.

V. Conclusions

We have found that LSTM networks provided the best results for automatic speech recognition of the Diné language. It produced consistently better results no matter which optimizer was used. It also took the least amount of time to train this model. We were also able to get good results with the CNN architecture. We are currently working on expanding our vocabulary and increasing our number of speakers.

Acknowledgment

Some of the material is based upon work supported by the National Science Foundation under Grant No. 1949230. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

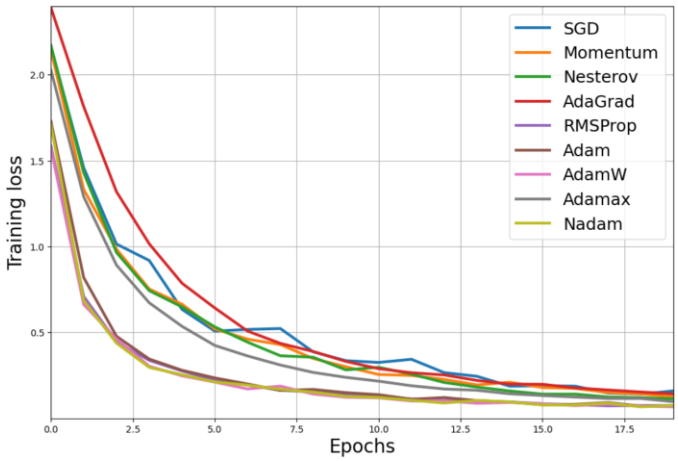

Fig. 2. LSTM Training Loss.

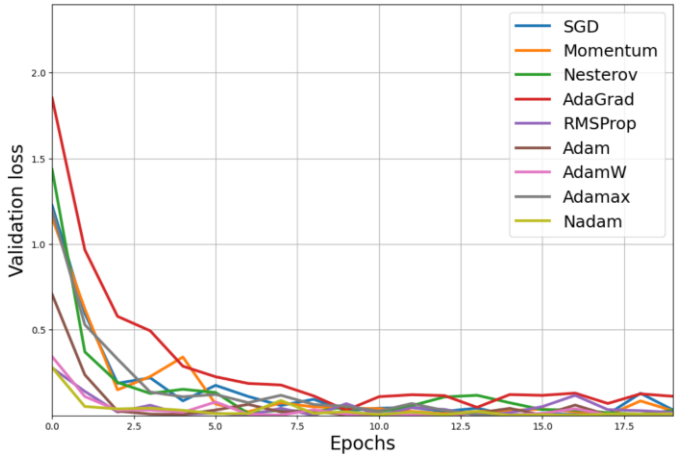

Fig. 3. LSTM Validation Loss

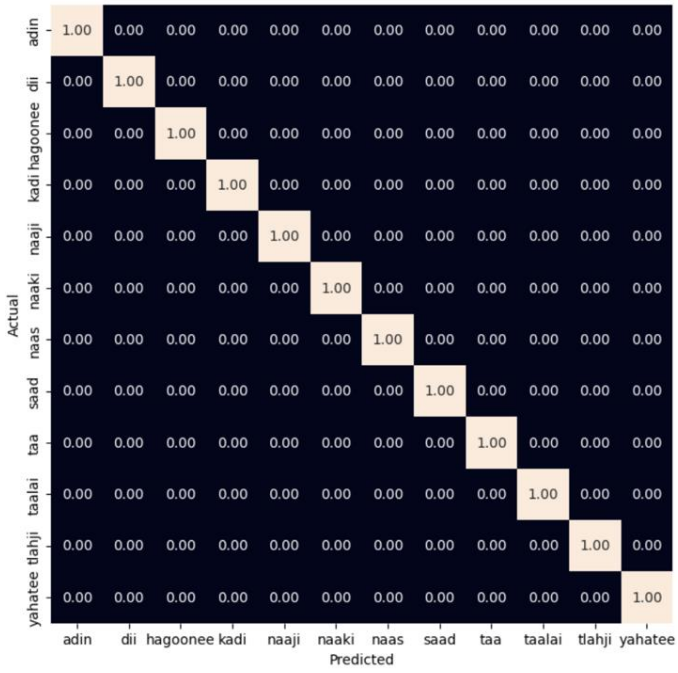

Fig. 4. LSTM model best result.

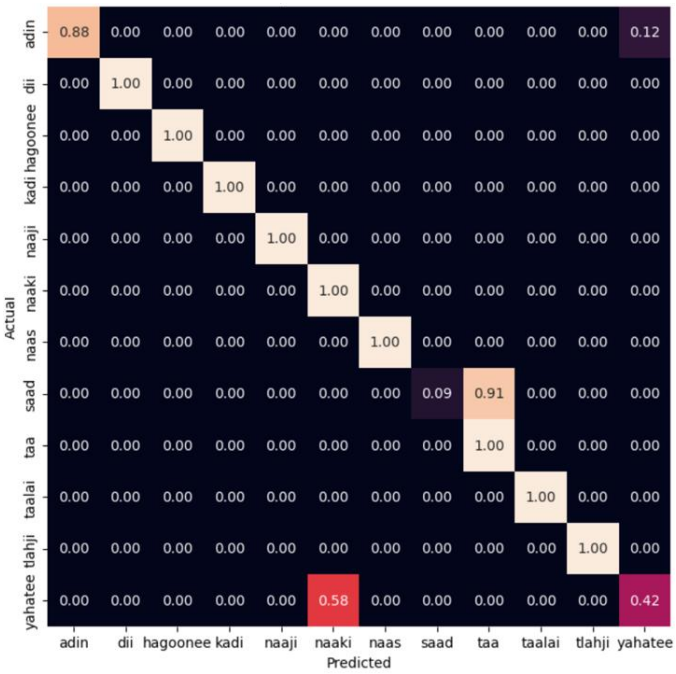

Fig. 5. LSTM model worst result

REFERENCES

[1] "Speech commands dataset version 2," [Online]. Available: http://download.tensorflow.org/data/speech_commands_v0.02.tar.gz, [Accessed: Dec 1, 2014].

[2] Linguistic data consortium, [Online]. Available: https://www.ldc.upenn.edu/, [Accessed: Dec 1, 2014].

[3] Mozilla common voice, [Online]. Available: https://voice.mozilla.org/en, [Accessed: Dec 1,2014].

[4] E. Parsons-Yazzie, B. Yazzie, T. Mcconnell, and B. Whitethorne, "The Navajo Alphabet and the Navajo Sound System," in *Diné Bizaad Bínáhoo'aah*, Flagstaff, Az: Salina Bookshelf, 2008, pp. 2.

[5] J. Minahan, "Navajos," in *Ethnic Groups of the Americas: An Encyclopedia*, Santa Barbara, California: ABC-CLIO, 2013, pp. 262.

[6] T. L. McCarty, "WE WERE TAUGHT ONLY BY ANGLOS," *A place to be Navajo: Rough Rock and the struggle for self-determination in indigenous schooling*, Erlbaum Publishers, Mahwah, NJ, 2002, pp. 51.

[7] G. Goertz, K. Laenté, S. Adams, J. Begay, M. Chee, "Use of causatives in Navajo: Syntax and morphology," 2006, [Online]. Available: https://www.linguistics.ucsb.edu/sites/secure.lsit.ucsb.edu.ling.d7/files/sitefiles/research/papers/18/Goertz.et_.al_._vol18.pdf

[8] R. Jimerson, Z. Liu, E. Prud'hommeaux, "An (unhelpful) guide to selecting the right ASR architecture for your under-resourced language," 2024, ACL Anthoology, [Online]. Available: https://aclanthology.org/2023.acl-short.87/

[9] A. Gomez, "Speaker Diarization and Identification From Single Channel Audio Recordings Using Virtual Microphones," *IEEE Access*, vol. 10, 2022.

[10] W. Shi, P. Tran, S. Celedón-Pattichis, and M.S. Pattichis, "Long-term Human Participation Assessment In Collaborative Learning Environments Using Dynamic Scene Analysis," *IEEE Access*, vol. 12, 2024.

[11] L. S. Tapia1, A. Gomez, M. Esparza, V. Jatla, M. Pattichis, S. Celedón-Pattichis, and C. LópezLeiva, "Bilingual Speech Recognition by Estimating Speaker Geometry from Video Data," in Computer Analysis of Images and Patterns: 19th International Conference, CAIP 2021, Virtual Event, September 28–30, 2021, Proceedings, Part I. Springer-Verlag, Berlin, Heidelberg, pp. 79–89. https://doi.org/10.1007/978-3-030-89128-2_8

[12] V. Jatla, S. Teeparthi, U. Egala, S. Celedón-Pattichis, M. Pattichis, “Fast Low-parameter Video Activity Localization in Collaboration Learning Environments,” 2024, doi: 10.1109/ACCESS.2017.DOI

[13] V. Jatla, S. Teeparthi, M. S. Pattichis, and S. Celedón-Pattichis and C. LópezLeiva, "Long-term Human Video Activity Quantification of Student Participation," *IEEE Access*, 55th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 2021, pp. 1132-1135, doi: 10.1109/IEEECONF53345.2021.9723241.

[14] W. Shi, M. S. Pattichis, S. Celedón-Pattichis and C. LópezLeiva, "Robust Head Detection in Collaborative Learning Environments Using AM-FM Representations," *IEEE Access*, IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), Las Vegas, NV, USA, 2018, pp. 1-4, doi: 10.1109/SSIAI.2018.8470355.

[15] W. Shi, M. S. Pattichis, S. Celedón-Pattichis and C. LópezLeiva, "Dynamic Group Interactions in Collaborative Learning Videos," 52nd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 2018, pp. 1528-1531, doi: 10.1109/ACSSC.2018.8645132.

[16] W. Shi, P. Tran, S. Celedón-Pattichis and M. S. Pattichis, "Long-Term Human Participation Assessment in Collaborative Learning Environments Using Dynamic Scene Analysis," in *IEEE Access*, vol. 12, pp. 53141-53157, 2024, doi: 10.1109/ACCESS.2024.3387932.

[17] Z. Jackson, 2016 , “Free spoken digit dataset (FSDD),” Kaggle, [Online]. Available: https://www.kaggle.com/datasets/joserzapata/free-spoken-digit-dataset-fsdd

[18] P Warden, “Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition,” 2018, arXiv:1804.03209. [Online]. Available: https://arxiv.org/abs/1804.03209

[19] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, “Audio Augmentation for Speech Recognition,” Interspeech 2015, pp. 3586-3589, doi: 10.21437/Interspeech.2015-711.

[20] A. Ragni, K. M. Knill, S. P. Rath, and M. J. F. Gales, “Data augmentation for low resource languages,” *Proc. Interspeech 2014*, [Online]. Available: http://mi.eng.cam.ac.uk/~mjfg/interspeech14-ragni.pdf

[21] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk and Quoc V. Le. “SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition,” 2019, arXiv:1904.08779. [Online]. Available: https://arxiv.org/abs/1904.08779