# A Functional NLP System for Twi (Akan) Using Limited Data

David Sasu
Ashesi University
Berekuso, Ghana
david.sasu@ashesi.edu.gh

Dennis Asamoah Owusu
Ashesi University
Berekuso, Ghana
dowusu@ashesi.edu.gh

## ABSTRACT

Natural Language Processing (NLP) continues to advance. However, native Ghanaian languages like many other languages in the world remain untouched. Limited availability of annotated data in these languages is, perhaps, the primary limiting factor. This work explores the development of a useful and practical NLP system for Twi, a native Ghanaian language, under the constraint of limited data. The result is a system developed using only 4.65 minutes of annotated speech data that allows searching for selected Twi songs by saying the song title in Twi.

## CCS CONCEPTS

• **Computing methodologies → Speech recognition**.

## KEYWORDS

natural language processing, speech recognition, low resource languages, information retrieval, twi, akan, ghana

**ACM Reference Format:**
David Sasu and Dennis Asamoah Owusu. 2018. A Functional NLP System for Twi (Akan) Using Limited Data. In *Accra '19: ACM SIGCAS Conference on Computing and Sustainable Societies, July 03–05, 2019, Accra, Ghana.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/1122445.1122456

## 1 INTRODUCTION

A major challenge in modern Natural Language Processing (NLP) is the development of systems that work well for low resource languages [6]. Low resource languages are languages without the scale of linguistic resources needed for the development of NLP systems. Native Ghanaian languages fall into this category. If NLP is to benefit as many people as possible, it is necessary to develop systems that overcome the data limitations of low resource languages.

An approach to tackle this challenge is the development of multilingual or cross-lingual models [3] [10]. Rather than training a model on a single language, the model is trained jointly on several languages. Transfer learning occurs, allowing the low resource languages to benefit from what the model learns from high resource languages. Another approach is to augment the limited data - to generate more data for training using the limited data [9] [7].

Semi-supervised learning and active learning are two other approaches [3] [12]. A language may have limited annotated data but unannotated data in abundance. Semi-supervised learning takes advantage of the unannotated data as well. Active learning enables efficient collection of training data. For instance, it could select more informative data for labeling instead of random labeling.

The hypothesis of our work is that, even with conventional NLP techniques, one can develop some useful and practical NLP system for low resource languages like native Ghanaian languages. The language chosen for this work is Twi - a dialect of Akan. The NLP system developed allows one to open the YouTube page of some selected Twi songs by saying the title of the Twi song[1]. It was trained with just 4.65 minutes of transcribed Twi speech. The results are very promising.

## 2 IMPLEMENTATION

The primary component of our song search system is an automatic speech recognition (ASR) system to recognize the song titles. 9 songs were chosen. The authors chose songs they knew to be popular. 9 recordings - one per song title - was obtained from 31 adult volunteers comprising 14 males and 17 females. The 279 recordings and their transcriptions constituted the training data for the ASR system. The ASR system was built with Kaldi [8] and based on Hidden Markov Models/Gaussian Mixture Models [14, p. 43].

### 2.1 Acoustic Model

The acoustic model of an ASR system learns statistical representations of the language sounds so that given an unknown sound, it can provide a probability distribution indicating which of the language sounds the unknown sound is likely to be. Phones or phonemes are the basic unit of sound in many ASR systems. To train a phoneme-based acoustic model using audio recordings and their transcriptions, a phonetic dictionary is used. The dictionary maps words to the sequence of phones that comprise the words.

Since there is no recognized phonetic dictionary for Twi, we improvised by looking up English words that sound like the Twi words of interest in an English pronunciation dictionary. CMUDict[2] was used. As an illustration, consider the Twi word "obiaa" which means "everyone". The "o" sounds like "o" in "old" so we look up "old" in CMUDict and get "OW L D ." as its pronunciation. We, thus, represent the "o" sound in "obiaa" as "OW".

Another important resource for developing the acoustic model is speaker identification information. Each audio must be associated with its speaker. In Kaldi, this is accomplished by creating a file with *utterance_ids* and *speaker_ids* and passing it to the *spk_to_spk2utt.pl*

*Accra '19, July 03–05, 2019, Accra, Ghana*

ACM ISBN 978-1-4503-9999-9/19/07…$15.00
https://doi.org/10.1145/1122445.1122456

[1] Video of a demo is available at https://youtu.be/6BLFVnv4LSE

[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

utility. Silence phones are also specified. Kaldi utilities are then used to generate additional data and put all data in the required format.

ASR systems do not, typically, work on raw audio signals such as what is contained in a .wav file. Features are extracted from the audio signal and used in model training and prediction. We used the Mel-frequency Cepstral Coefficients (MFCC)[4] as features. Extracting MFCC features from the audio yields a sequence of frames (vectors) for each audio file. Speaker based Cepstral Mean and Variance Normalization is applied [13]. The Kaldi utility *steps/train_sat.sh* was used to train the acoustic model. It trains on fMMLR-adapted features [5].

### 2.2 Language Model

An ASR, typically, uses a language model to transcribe audio at test time. A bigram model was generated from a text corpus comprising the 9 selected Twi songs. The model was generated in ARPA format using the SRI Language Modeling toolkit [11].

### 2.3 Song Search

Apart from the ASR system to transcribe the audio, the other component needed for our system is the song search. All 9 songs and their corresponding YouTube links were stored in a text file. The search component takes the best transcription from the ASR system and searches for the title most similar to the transcription. The Ratcliff-Obershelp algorithm was used for computing similarity [2]. It was implemented using the python difflib library [1].

## 3 RESULTS

Intrinsic and extrinsic evaluations were performed. The intrinsic evaluation looked at the Word Error Rate (WER) of the ASR system. Two experiments were performed for intrinsic evaluation. In experiment 1, the ASR system was built using audio from only one speaker and then evaluated by letting the same speaker say Twi song titles. The WER was 0.0. In experiment 2, the ASR system was built using all the collected audio and then evaluated by asking new speakers to say the Twi song titles. The WER was 0.216.

The extrinsic evaluations looked at whether the system played the song the user requested. In the first extrinsic evaluation, a volunteer who provided speech for training used the system to request different songs 10 times. The system worked 95.4% of the time. In the second extrinsic evaluation, 10 volunteers who did not provide any speech data for training used the system to request different songs 10 times. The system worked 94.3%.

## 4 CONCLUSION AND DISCUSSION

Our results show that a functional NLP system can be developed for low resource languages like Twi using limited data. Trained using only 4.65 minutes of speech data and transcriptions, the system was able to retrieve a requested song title 94.3% of the time even when the requester's voice was not used in training.

The system is limited in that it only recognizes words that were used during training. However, even with this limitation, it is still a practical and useful system that allows users to search for songs. It can likely be extended to include many other songs by collecting a similarly limited amount of speech data. This approach could have applications in other contexts as well.

Consider the development of a voice user interface system that allows illiterate farmers to request, in Twi, the current prices of their goods or what tomorrow's weather will be like and other such essential information. The variation in sentences and words used in such a system will be fairly limited and this work suggests that such a voice interface system could be developed by collecting a modest amount of speech data.

We also suspect that the system could recognize, fairly well, words whose phonemes were seen in training even if the words themselves were not. In future work, we plan to extend the language model to include words not included in the speech training data but whose phonemes are present.

It is important to note that the system as a whole performed better than the ASR component. This is because even if there was an error in transcription, the similarity algorithm can, often, still find the requested song. This demonstrates synergy that can be explored to develop NLP systems for low-resource languages where the paucity data limits the performance of the individual components.

## ACKNOWLEDGMENTS

We acknowledge the help provided by Constant Likudie at Ashesi University as well as support from the Computer Science department of Ashesi University.

## REFERENCES

[1] [n. d.]. Difflib. Retrieved May 16, 2019 from http://www.poker-edge.com/stats.php

[2] Paul E. Black. 2004. Ratcliff/Obershelp pattern recognition. In *Dictionary of Algorithms and Data Structures [Online]*, Paul E. Black (Ed.). https://www.nist.gov/dads/HTML/ratcliffObershelp.html

[3] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, M. Picheny, Z. TÃijske, P. Golik, R. SchlÃijter, H. Ney, M. J. F. Gales, K. M. Knill, A. Ragni, H. Wang, and P. Woodland. 2015. Multilingual representations for low resource speech recognition and keyword search. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. 259–266. https://doi.org/10.1109/ASRU.2015.7404803

[4] Steven Davis and Paul Mermelstein. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing* 28, 4 (1980), 357–366.

[5] Mark JF Gales. 1998. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer speech & language* 12, 2 (1998), 75–98.

[6] Julia Hirschberg and Christopher D. Manning. 2015. Advances in natural language processing. 349, 6245 (2015), 261–266. https://doi.org/10.1126/science.aaa8685

[7] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. *arXiv e-prints*, Article arXiv:1904.08779 (Apr 2019), arXiv:1904.08779 pages. arXiv:eess.AS/1904.08779

[8] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. 2011. The Kaldi Speech Recognition Toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.

[9] A. Ragni, K. M. Knill, S. P. Rath, and M. J. F. Gales. 2014. Data augmentation for low resource languages. In *INTERSPEECH*, Vol. 2.

[10] Sebastian Ruder. 2017. A survey of cross-lingual embedding models. *CoRR* abs/1706.04902 (2017). arXiv:1706.04902 http://arxiv.org/abs/1706.04902

[11] A Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of ICSLP*, J. H. L. Hansen and B. Pellom (Eds.), Vol. 2. Denver, 901–904.

[12] A. R. Syed, A. Rosenberg, and M. Mandel. 2017. Active learning for low-resource speech recognition: Impact of selection size and language modeling data. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 5315–5319. https://doi.org/10.1109/ICASSP.2017.7953171

[13] Olli Viikki and Kari Laurila. 1998. Cepstral domain segmental feature vector normalization for noise robust speech recognition. *Speech Communication* 25, 1 (1998), 133 – 147. https://doi.org/10.1016/S0167-6393(98)00033-8

[14] Dong Yu and Li Deng. 2016. *AUTOMATIC SPEECH RECOGNITION.* Springer.