Mobile Application for Improving Speech and Text Data Collection Approach

Sarah Samson Juan[1] and Jennifer Fiona Wilfred Busu[1,2]
[1]*Institute of Social Informatics and Technological Innovations, Universiti Malaysia Sarawak, Sarawak, MALAYSIA.*
[2]*Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Sarawak, MALAYSIA.*
sjsflora@unimas.my

***Abstract*—This paper describes our work in developing a mobile application for collecting language speech and text data. The application is built to assist linguists or researchers in simplifying their tasks in data collection who of native speakers living in remote interiors. Researchers rely on numerous apparatus to carry out their tasks to capture audio or text from far to reach places, but with this mobile application, they would only need to carry one device, which can ease their logistics troubles. The mobile app, named as Kalaka, is designed for users to store details of native speakers, record speech and insert speech transcripts all in one platform. Kalaka is built on the Android platform, which allows data stored in the mobile device to be transferred to a cloud storage using WiFi networks. Usability tests performed in respondents shows, all participants in the evaluation are able to use the application to record their voices and save texts. We also received positive feedbacks on the mobile application from our survey, with more than half of the respondents gave their confidence using Kalaka and they would use the system frequently.**

***Index Terms*—Mobile Application; Data Collection Tools; Corpus Development.**

I. INTRODUCTION

Language documentation is a process of recording linguistic properties which could help in preserving an identity of a language. This process is tedious for linguists and researchers. They may need to travel to rural areas to collect data from native speakers. Most of the time, researchers may use several tools for recording and transcribing speech data such as speech recorder, papers, or computers. However, keeping speech and transcripts in separate tools (recorder and notebook, for example) during a data collection trip has high risks. Researchers could lose valuable data if one or both tools broken or damaged. Moreover, storing or labelling data poorly could cause data loss.

Mobile devices can store audio and digital text data. They are lightweight and easy to carry around when traveling in rural areas. Nowadays, open source operating system such as Android OS, enables developers to create simple applications for low-cost mobile devices. This is cost-effective for researchers who are constantly on the move and need to gather a lot of data from native speakers in rural areas with limited connectivity. Furthermore, with the availability of WiFi and cloud technologies support in mobile platform, developers can build an application for sharing data to cloud storage or real-time database. Hence, researcher can do a backup whenever it is necessary.

Thus, this paper reports our first steps in developing an open source mobile application called *Kalaka*, which can store speech and digital text. Among other functionalities of the application are, store speaker details, edit entries to list of languages, categories or origins, and synchronize data from device to cloud. The latter can be done when a WiFi network is available.

The flow of the paper is as follows. In Section II, we describe our motivation for building creating digital tools to collect speech data and Section III explains the development of *Kalaka* for mobile devices. Subsequently, Section IV presents the interfaces of the system and Section V reports results of our usability evaluation. Finally, Section VI concludes our paper and describes our next steps.

II. COLLECTING DATA FOR RESEARCH IN SPEECH TECHNOLOGY

In speech technology, researchers have developed many methods for building speech applications that could help in human-computer interaction. For example, automatic speech recognition (ASR) systems are used to convert human speech to readable texts. Currently, there are many well-known ASR systems such as Apple's Siri, Microsoft's Cortana and Google's Google Now.

To build ASR, the following data are needed [1-3]:

i) Speech and speech transcripts for acoustic model – typically, a minimum of 20 hours of speech from female and male speakers of each language
ii) Text for language model – a minimum of 500MB of text for training a language model. The language model is used to define the grammar rule and it helps to select the best ASR output.

The presence of Sarawak languages in ASR research is still very low. Juan [4] has published a thesis on exploring resources for building ASR for under-resourced languages in Malaysia. The author's work focused on developing ASR for Iban, a language that is largely spoken in Sarawak. The Iban ASR was built using 7 hours of transcribed speech and text data with 2 million words. Due to inconsistent spellings found in Iban text and very low amount of transcribed speech, the Iban ASR achieved 85% accuracy [5]. This performance is considered low compared to other state-of-the-art ASR systems ([6] - 90-95% accuracy).

Thus, doing research in Speech Technology in Sarawak languages is a challenge as we need to collect large amount or data for building speech applications. There are several related issues such as:

i) Native (original) speakers live in rural areas
   - Travelling cost is high for researchers to meet native speakers
ii) The low amount of electronic text data available
   - Not many digital texts in target language.

Digitizing hardcopy materials could be one of the ways to collect texts, but this process is also time consuming.

iii) Lack of unique or stable orthography system
- Not many Sarawak languages have a standard orthography system. ASR performance can be affected by inconsistent spellings in training data

iv) Poor management of acquired data
- Loss of data due to mismanagement of data collection tools

There has been an increasing number of digital applications that are used for language maintenance and revitalization. The SIL International and Payap Language Software produced a software for creating dictionaries. This software is called WeSay (available at: www.wesay.org) is a free software that can be used by non-linguists to build dictionaries in their own languages [7].

An e-dictionary for 16 indigenous languages in Taiwan was created by Taiwan's Indigenous Council with the aim to standardize the orthography systems of these languages. The contents of the online dictionary are based on digitized printed materials [8]. *Zahwa*, a mobile application, enables users to take photos and short videos [9]. Furthermore, it allows users to do voice-over recording while swiping through the photos and videos. The developers designed the application for documenting recipes or any procedural discourse. Users can record instructions in their own languages and provide translations.

Bird et al. [10] developed a mobile application as a platform for collaborative language documentation. The app, called *Aikuma*, can store bilingual speech data, where users can record, re-speak and do oral translations. The data which are stored in a mobile device can be retrieved from its browser-based transcription tool, which can be connected through WiFi, Bluetooth or USB interfaces.

Our project aims to build a mobile application that can store and organise both speech and text data. The application can also send the acquired data to a cloud storage. The following section explains the requirements and design of the system.

III. FUNCTIONALITIES OF *KALAKA*

We take into several considerations in the design of the application. Figure 1 illustrates the users and functionalities of *Kalaka*. There are three types of app users; community or native speakers, an administrator who is responsible of the device and super administrator who is a project leader who can control all devices with *Kalaka*.

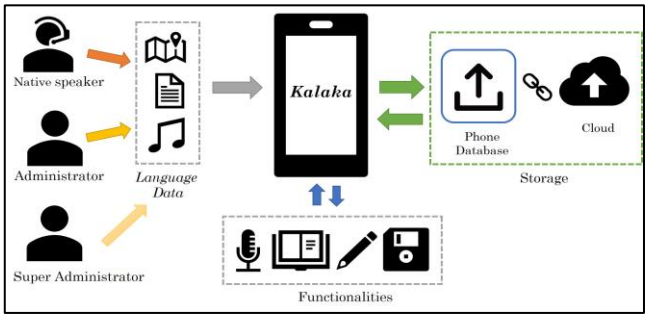


Figure 1: Overview of the functionalities of *Kalaka*

Community or native speakers can keep speaker's details such as origin (location), language spoken, name, age, gender and phone number. They can also choose the category of their recordings, such as food, home, education or agriculture. Moreover, the application allows them to transcribe their recordings. Both data will be stored in the device after each session is saved.

The administrator will be able to edit information in the device through the application such as; remove speech data and registered speaker details or edit origin list, category list and language list. Prior to performing these tasks, the person is required to login his details in the application.

The super administrator or project leader can synchronize selected information across devices with *Kalaka* as well as uploading recordings and annotations to cloud storage. The project leader needs to log his details into the system and get connected to a WiFi network to carry out the tasks.

IV. IMPLEMENTATION

Kalaka is developed for Android devices, as our first intention is to build an open source application that can be used in low-cost mobile devices. The application should be simple and has a user-friendly interface, which can be easily used by untrained users for collecting data.

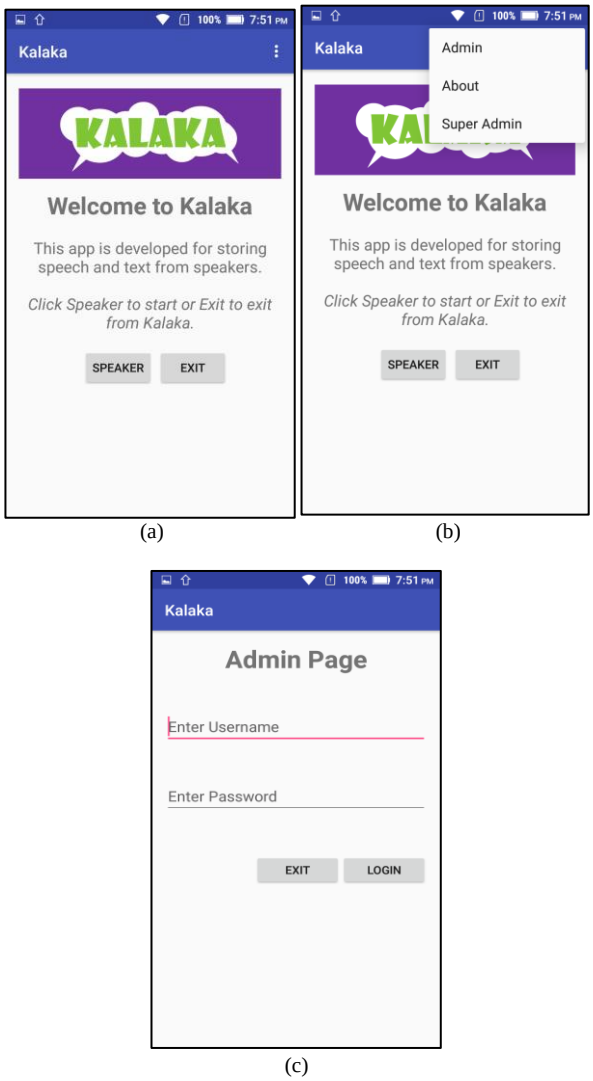
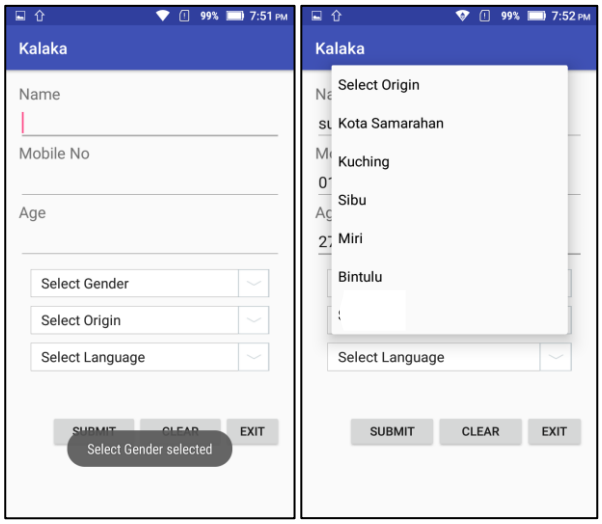


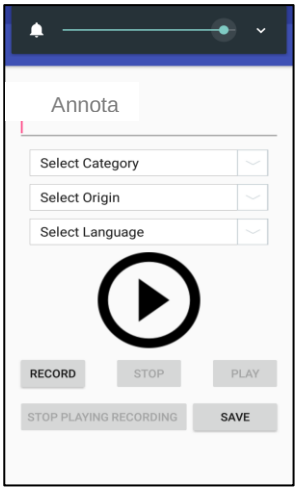
Figure 2: Screens for welcome and login for administrator.

Figure 2 (a) shows the welcome page of *Kalaka*, Figure 2(b) and (c) show steps to login page for administrator. In the

welcome screen, native speaker can access the registration page by clicking the "Speaker" button, while administrator or super administrator will need to press the top right tab to choose their type of login.



(a) (b)

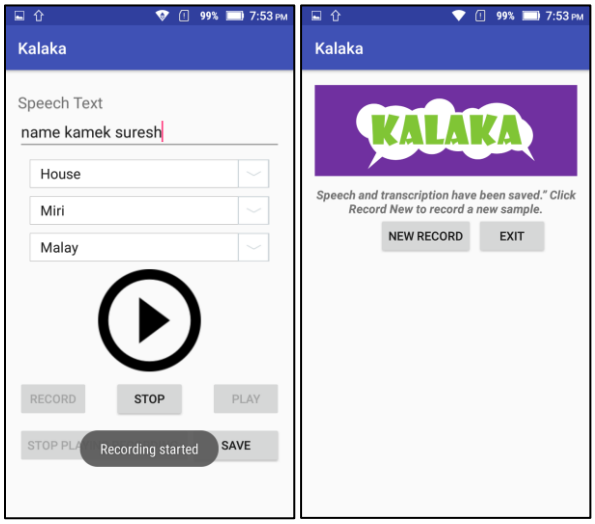


(c)

Figure 3: Screens for registration, recording and annotate speech.

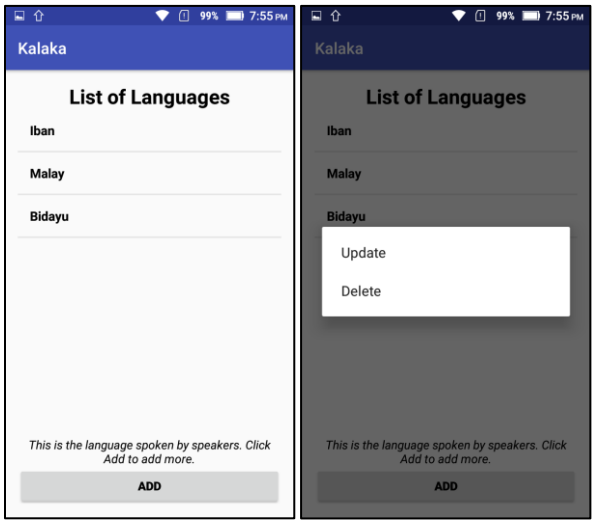
At the registration page, the speaker is required to complete details as shown in Figure 3 (a). In Figure 3(b), we show the drop-down list for the speaker to select the origin or the location of the speaker. Besides list of origin, we also provide list of languages and gender. Once registration is complete, the speaker clicks "Submit" to access to the recording page, as shown in Figure 3 (c). In this page, the speaker is required to select category, language and speaker's location.

Figure 4 (a) displays an example of a recording session while Figure 4(b) shows a confirmation page after the speaker saves the speech and text data in the device. In this page, the speaker can choose to quit or continue recording more data. When the administrator page is accessed, the person can update information such as remove speech data, edit category, language and origin. Figure 5 (a) shows the list of languages registered in the application and Figure 5(b) shows a pop-up screen to update or delete one of the entries. This function is necessary to correct any an error that has been entered.



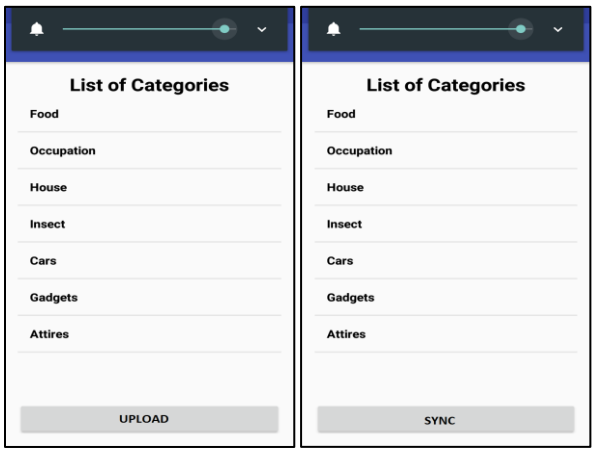
(a) (b)

Figure 4: (a) Recording speech with transcription added (b) After saving speech and its transcription, a page to confirm that the data has been saved will appear.



(a) (b)

Figure 5: (a) Entries in the language list (b) A pop-up screen to update or delete an entry.



(a) (b)

Figure 6: (a) Upload and (b) sync buttons at the super administrator page

At the super administrator page, the project leader can upload data in the mobile device to a cloud or synchronize selected information from the cloud to the device. To build

these functions, we use Firebase https://firebase.google.com/) as the backend to upload and synchronize data. The following Figure 6 (a) and (b) show the super administrator's pages, which are slightly different than the administrator's pages. In the next section, we report the results of the usability evaluation on *Kalaka*.

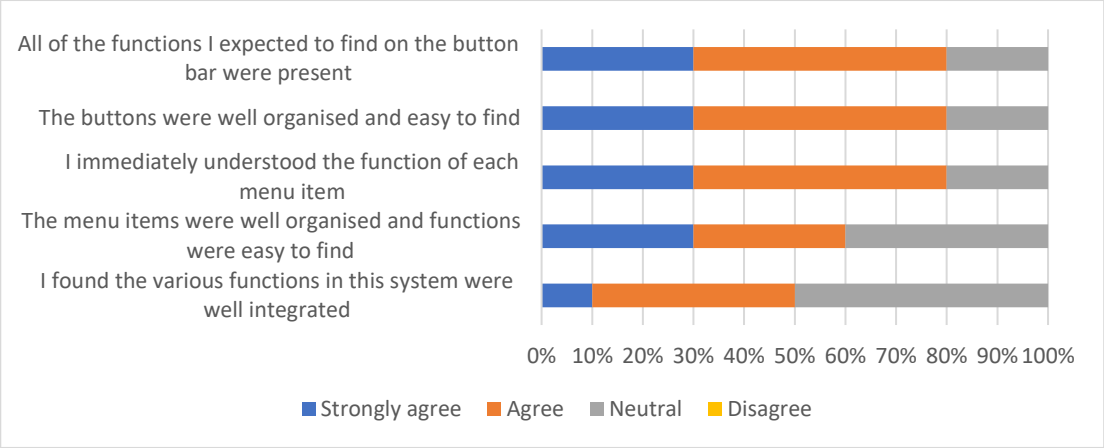


Figure 7: Users' feedback on the functionalities of Kalaka

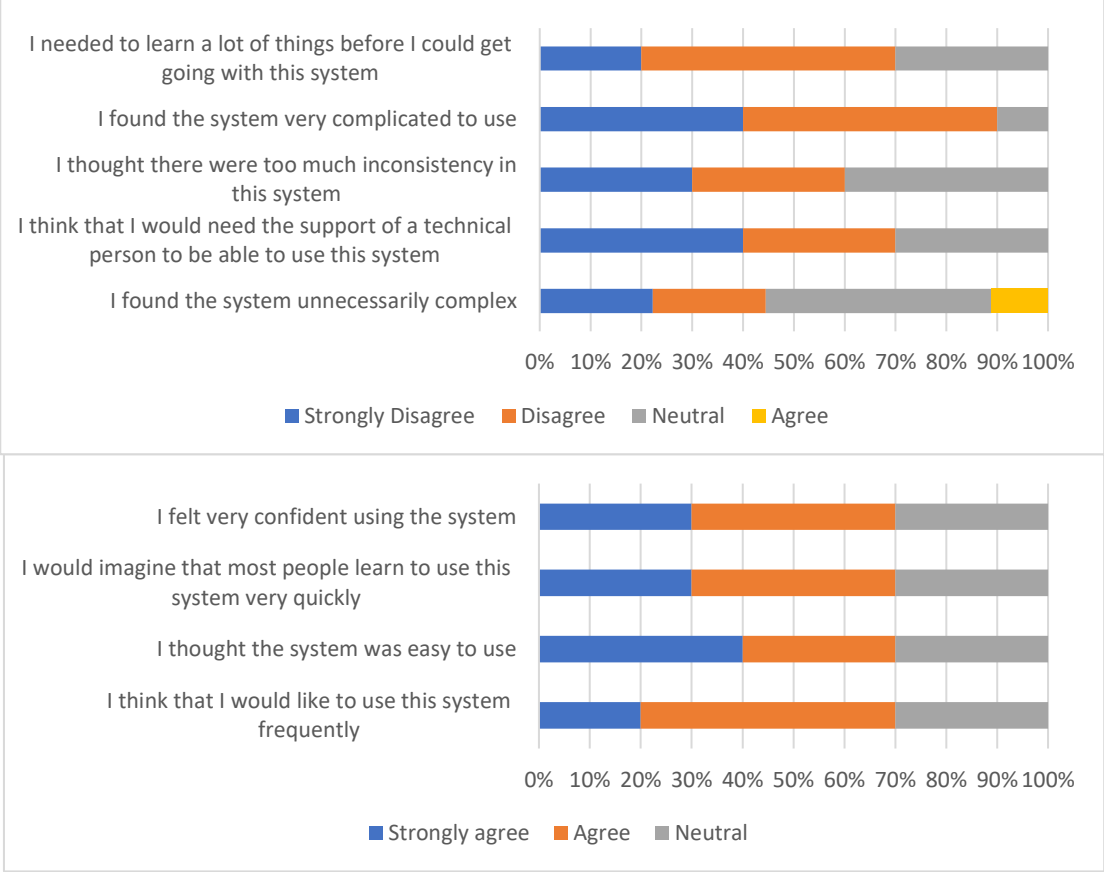


Figure 8: Users' experiences on using Kalaka

V. Usability Testing

We prepared simple tasks for users to use *Kalaka* for conducting a usability evaluation. Each user is required to install the application in an Android device, register his details in the system, choose category and insert a text and finally, record speech by reading text and save the recording. Ten people participated in the evaluation and shared their experiences by completing our survey. All of them have successfully completed the tasks and recorded 10 sentences in Sarawak Malay, English, Iban and Bidayuh languages.

The results of the survey are as shown in Figure 7 and Figure 8. To check the functionalities of Kalaka, we asked five questions to each of the participants. As shown in Figure 7, at least half of them agreed that the buttons are well organised and integrated. More than half of them understood the functions available in *Kalaka* and agreed that all functions that they expected were visible. Figure 8 shows participants' opinions about the system. Overall, we received positive feedbacks regarding the application. Most of them found that the system was not difficult to use and felt confident using *Kalaka*. Besides that, they would like to use the system frequently. Surprisingly, one of them found the system unnecessarily complex.

VI. Conclusions and Future Work

In this paper, we demonstrated our work on building a digital transcription tool for mobile devices called *Kalaka*. In coming up with the design and functionalities of the system, we take into several considerations. Native speakers should be able to use the application with minor or no trainings, data collector or administrator of the device can modify data on the device and the project leader has the authority to synchronize data for devices with *Kalaka* using a real-time database. We have successfully implemented our idea using Android OS and have presented several interfaces of the mobile application. We also performed a usability evaluation to test the functionalities and collect feedbacks from users. Overall, our users successfully stored their details and recorded their voices using *Kalaka*. Furthermore, the application was showcased for the first time at the recent UNIMAS Innovation and Technology Expo (InTEX 2017) and our product obtained bronze. Our further work is to perform tests on synchronizing data from the device to Firebase and using the mobile application to collect Sarawak languages from several locations.

Acknowledgements

This work is supported by Universiti Malaysia Sarawak under Special Research Fund grant no. I03/SpFRI/1429/16/6. We would like to thank Suresh Ramachandran for his technical work of this project.

References

[1] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," In *Conference Proceedings of IEEE*, vol. 77, pp. 257-286, 1989.

[2] T. Schultz, *GlobalPhone: A multilingual speech and text database developed at Karlsruhe University*, pp. 345-348, 2002.

[3] L. Besacier, E. Barnard, A. Karpov, and T. Schultz, "Automatic speech recognition for under-resourced Languages: A Survey," *Speech Communication Journal*, vol. 56, pp. 85-100, Jan. 2014.

[4] S. Juan, *Exploiting resources from closely-related languages for automatic speech recognition system for low-resource languages from Malaysia*, Grenoble, France: Université Grenoble-Alpes, 2015.

[5] S. S. Juan, L. Besacier, B. Lecouteux, and M. Dyab, "Using resources from a closely-related language to develop ASR for a very under-resourced language: A case study for Iban," In *INTERSPEECH*, Dresden, Germany, 2015.

[6] G. Boulianne, L. Burget, A. Ghoshal, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, D. Povey, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi speech recognition toolkit," In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, Hawaii, 2011.

[7] E. Albright and J. Hatton, "Wesay, a tool for engaging communities in dictionary building," In V. D. Rau and M. Florey, eds., *Language Documentation and Conservation Special Publication No. 1: Documenting and Revitalizing Austronesian Languages*, p. 189201. University of Hawaii Press, 2008. Available at: http://hdl.handle.net/10125/1368.

[8] Taiwan Indigenous Council, Aboriginal Ethnic Language Dictionary, 2016: http://e-dictionary.apc.gov.tw/Index.htm

[9] M. Bettinson and S. Bird, "Developing a suite of mobile applications for collaborative language documentation," In *Workshop on the Use of Computational Methods in the Study of Endangered Languages*, Honolulu, 2017. Available at: http://www.aclweb.org/anthology/W/W17/W17-0121.pdf

[10] S. Bird, F. R. Hanke, O. Adams, H. Lee, "Aikuma: A mobile app for collaborative language documentation," In *Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pp. 1-5, Baltimore, USA, 2014.