Cloud-based speech technology for assistive technology applications (CloudCAST)

First AUTHOR[a,1] and Second AUTHOR[b]
[a] *Affiliation*
[b] *Affiliation*

Abstract. The CloudCAST platform provides a series of speech recognition services that can be integrated into assistive technology applications. The platform and the services provided by the public API are described. Several exemplar applications have been developed to demonstrate the platform to potential developers and users.

Keywords. Keyword, keyword

1. Introduction

In this paper we introduce the CloudCAST platform, which is the product of a Leverhulme Trust International Network funded from January 2015 for 3 years. The network partners are. Anonymous in the draft

In recent years there has been considerable research into clinical applications of speech technology (CAST), particularly approaches for assessing disordered speech [1], supporting pronunciation training [2], recognition of disordered speech [3] and voice reconstruction [4].

The aim of the CloudCAST network was to produce a platform that will make it easier for clinical applications using speech technologies to be developed and implemented. Through it assistive technology providers, and professionals working with clients with speech and language difficulties, including therapists, pathologists, teachers, and assistive technology experts, will be able to use tools to collaborate and to develop targeted speech recognisers.

The CloudCAST platform is a cloud-based service that is available to users worldwide. Where possible the tools that are provided are free to use, and can be deployed in applications that require personalised speech recognition, speech-based diagnosis and interactive spoken language learning. The platform provides interfaces which will make these tools easy to use for professionals, who are not necessarily speech technology experts, and their clients.

The platform can also be used by speech technologists to facilitate the collection of speech data required for the machine learning techniques that underpin modern speech technology. A problem for field of clinical applications of speech technology is that

[1] Corresponding Author.

there is little training data in comparison to that available for mainstream technology development. To address this problem, and subject to appropriate ethical consent, it will be possible for interactions with CloudCAST to be recorded. Thus the data that is collected can be used to retrain and improve the performance of the speech recognition over time.

We also provide access to existing databases of speech that developers may wish to use in their applications. In this way we hope by the end of the project, to create a self-sustaining platform with a developer community who will be able to continue CloudCAST development beyond the end of our current funding period. We are working with speech technologists in research laboratories in several countries to make their software and data available to the users of the CloudCAST platform.

With the overarching aim in mind, CloudCAST provides interfaces, resources and tools targeted at several types of users:

- Application developers, who want to embed CloudCAST services into their own applications, for instance voice control of domestic robots.
- Speech professionals such as speech and language therapists, who want to use CloudCAST applications to work with their clients, for instance, to devise personalised therapy exercise programmes.
- Assistive technology professionals who may wish to integrate speech technology into assistive applications and services such as for environmental control. Because physical disabilities often co-occur with speech disorders, off-the-shelf technology is likely to perform poorly in such applications.
- End-users of these different applications, so that they can manage the data that is stored about them.

In this paper we describe the problems the platform solves and the architecture and API that we have implemented. We also describe the some exemplar applications we have developed to demonstrate the potential of the platform.

2. The challenges for CloudCAST

The CloudCAST platform has been designed to meet a series of challenges we identified as being crucial to making it attractive and useful to developers and end-users alike. Theses challenges were:

1. **To provide a responsive and adaptable platform**. A cloud-based solution is required to be able to make the platform responsive to users no-matter where they are located in the world. There is also a considerable literature that shows that for people with speech disorders speech recognition technology works best when it can be personalised to match the characteristics of the individual. Therefore to be able to maximise the benefits of speech technology for the potential end-users it was vital to provide the ability to train and adapt recognisers to individual users.

2. **Be robust to varying equipment and environments**. A constant challenge for speech recognition is the quality of microphones and the varying acoustic environments from which the speech originates. The end-users of CloudCAST-based applications will use a variety of equipment and be working in different environments, so the

CloudCAST speech recognition tools must be sufficiently robust to these factors.

3. **Provide reusable and scalable tools**. To make the tools most attractive to developers without expertise in speech technology they must be accessible and, ideally, reusable in simple ways. Therefore, new developers should be able to work with existing applications or extend existing tools for their own purposes.

4. **Store data securely**. A cloud-based solution implies that speech data will be transferred from end-users to a CloudCAST server for processing. Once processed that data could be stored, with appropriate user permissions, to enable developers to improve or adjust speech models over the course of the users' interactions with an application. This requires an appropriate and secure database for storing speech data.

5. **Have a flexible development architecture**. It is intended that developers should be able to use the platform to develop any applications which require speech technology, with a particular focus on people who may have a speech disorder or other similar need such as children learning to read. Therefore the architecture of the platform must be sufficiently flexible to support any potential interactions based on the speech tools that are provided.

6. **Include demonstrator applications**. To foster a community of developers who are motivated to use the platform we have identified a series of demonstrator applications to show how to use the platform and the potential power of the underlying speech technology. Two of these exemplars will be described in this paper.

3. The CloudCAST platform

There are now several different cloud-based speech recognition services available to developers who want to incorporate speech input into their applications. These services, such as Alexa Voice Service [5] and the Google Speech API [6], provide reliable speech recognition but crucially lack the customisation required for many assistive technology applications. This means applications for environmental control, which would greatly benefit users with physical impairment, are hard to develop because their kind of impairment often affects their speech, rendering these services unusable (see [7]). Likewise, the lack of flexibility regarding the output obtained and the level of detail allowed limits the kind of applications that can be built using these services. Crucially the suitable approach to using speech recognition for people with speech disorders, adapting or training personalised models, is not available via existing cloud-based services.

In contrast, CloudCAST aims to provide developers the widest possible control of the speech technology they wish to use for the creation of their applications, but without requiring expertise in this specialist technology.

The platform provides speech recognition services and allows developers to tailor the recognisers and their output to suit the specific needs of their end-users. Developers can interact with the service through an established API. This API allows the developer to generate datasets from both data collected through their application and that contributed to the CloudCAST platform to train and adapt recognition models. This approach will allow developers through their applications to provide, where necessary, personalised recognisers to meet the need of their end-users.

Developers are able to register applications with the CloudCAST platform, which allows the service to link specific requests to the external applications they came from.

Within CloudCAST, all applications are created by a single user (the developer), but an application can have multiple users (both professional users and the end-users). Most calls made to the CloudCAST server will be made from the context of an application, and users will only be able to make those that are available to their specific role.

The recognition service itself is built on top of the Kaldi speech recognition toolkit [8], a state of the art solution licensed as free-software, with a modular design and an active developer community. Interaction with Kaldi is done using GStreamer, using a design that was based on Tanel Alumae's kaldi-gstreamer-server [[Alumae2016]], but modified to suit the specific needs of the CloudCAST project. In particular, this design makes it possible to separate the handling of incoming connections, a task of the server, from the processing of the speech data, which is delegated to separate worker processes ensuring the scalability of the project.

The interaction between applications using the CloudCAST platform and the server processes is shown in Figure 1. In the diagram the exemplar applications described later in this paper are shown and their interactions with the server and the worker processes.

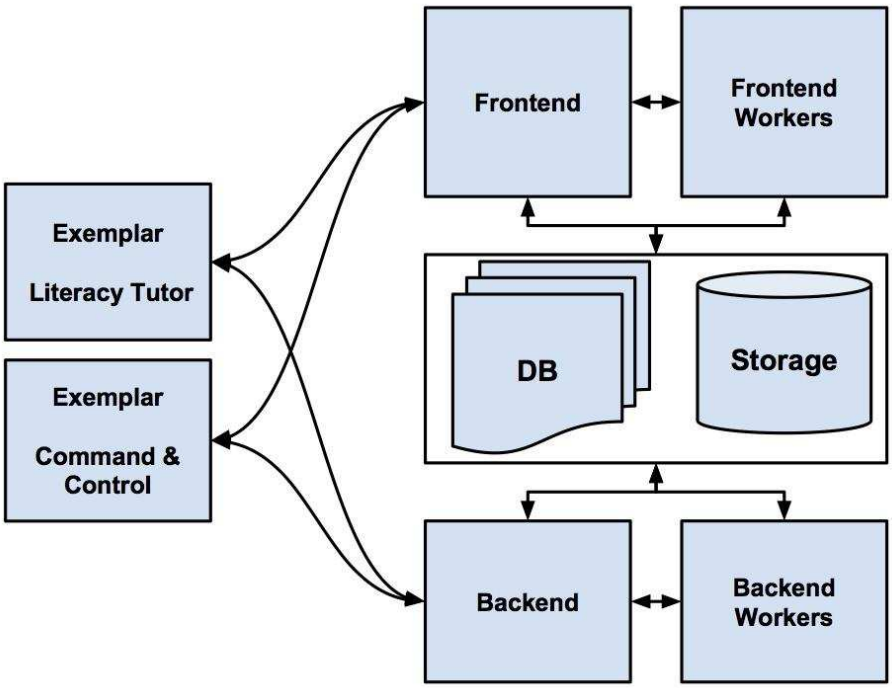

Figure 1. Schematic diagram of the CloudCAST platform and example applications.

4. Data collection and management

CloudCAST will also serve as a data repository for the distribution of existing databases and for the acquisition of new databases, and provide tools for data collection. Below we discuss the first database that will become freely available in CloudCAST, TORGO, and the database scheme we will use to represent future data collection

4.1. The database scheme

New users of CloudCAST can immediately use our database framework for representing the data their applications collects. To a large extent, this framework is designed to be generic to all speech recording tasks, and not all components need to be utilised. The database schema is broken down into three core sections: the subject, the task, and the session. A high-level overview of the data representation is shown in figure 2.

The subject section generally involves aspects related to the speaker, including demographics, levels of permission to use the data, and factors affecting the subject's language quality, such as country of origin, country of residence, spoken languages, history of smoking, and education level. The task section specifies the language task (e.g., picture description, conversation, reading of text, repetition of audio) along with a bank of available task instances (e.g., pictures to be used in the picture description task). The system supports a variety of question and answer types, including text, speech, multiple-choice, and fill- in-the-blank, with the ability for easy extension to new types. Each task instance is optionally rated with a level of difficulty, measured across arbitrary dimensions (e.g., phonological complexity, syntactic complexity). Information related to automatic scoring of tasks is stored along with each task instance, where appropriate (e.g., the correct answer to a multiple-choice question). Each subject can be associated with a number of recording sessions, and each session can be associated with a number of task instances. The session section stores the subject responses to specific task instances every time they interact with the system. This includes their language data, as well as meta- data such as total amount of time spent on each task, and date of completion. This database is designed to be extensible to future needs, and will be especially useful to streamline data organization to projects that otherwise have a more clinical focus. It enables:

1. longitudinal subject assessments, due to the ability to accommodate multiple language task instances in order to avoid 'the learning effect' over time,
2. dynamic variation of task instance difficulty and type based on subject performance, and
3. automated scoring of subject performance where appropriate.

4.2. Ethics and security

Use of the CloudCAST platform will necessarily involve the transmission and optional storage of speech data of end-users. Therefore we have designed our processes to ensure the security of that data, and given end-users full control over what data is retained on the CloudCAST platform. As part of this process professionals who initiate a service through CloudCAST will need to first confirm that they are abiding by the local ethics and governance rules.

For individual participants making use of CloudCAST services we will follow a process approved by the University of Sheffield Research Ethics Committee. It is proposed that as part of this process we will first fully explain to each individual user when they register with CloudCAST the background to the project and how we intend to use their speech data. Participants will be able to opt-in to different levels of engagement with the CloudCAST initiative. At the most basic level, a participant will be able to make use of the CloudCAST services without their data being used for further research, or shared with other researchers. The second level of participation can be selected by the participant when they wish to allow the CloudCAST team to retain their data for further research. The final level of participation can be selected by participants when they wish their data to be retained and potentially distributed to other speech researchers. As part of the on-going relationship with the participants, they will periodically be asked to re-confirm their consent for their data to be used in the way they chose.

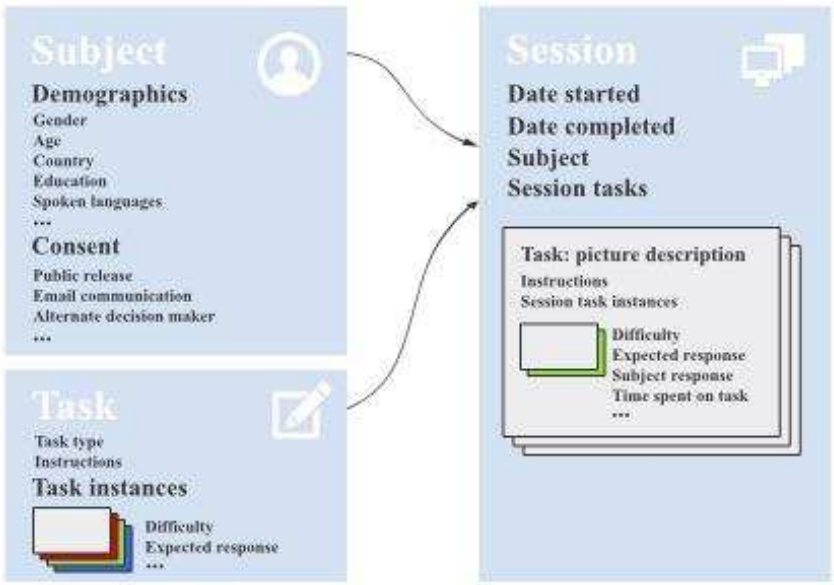

Figure 2. Simplified overview of database tables.

5. Exemplars

As part of the initial release of the CloudCAST platform we have implemented several example applications, which demonstrate some of the facilities of the platform.
We think these examples will not only be useful for professional and lay clients in their own right, but also will showcase to developers what the CloudCAST platform can be used for. In the spirit of an open source approach, these examples will be available for developers to adapt for their own needs.

5.1. Environmental control

An example of how to use the CloudCAST for a command and control application can be found in an environmental control application described in a companion paper presented in this session.

5.2. Reading tutor

An automated literacy tutor has been developed that will showcase the flexibility and overall utility of the tools provided. Speech-enabled literacy tutors have been developed and deployed with some success (see [10] for a brief overview). These tools, however have limitations relating to accessibility and flexibility.

The tool has been developed for use by Jamaican children learning to read English as a first language and by bilingual students in Italy, to practice reading aloud in English. A user-centred design approach was followed to produce the exemplar and we have engaged in a number of interactive sessions with intended user groups. Consultations with Italian speech and language therapists and teachers of English have also taken place.

The tool provides for reading exercises to be completed by the students (who are the end-users of the application). Teachers of the students (the professional users of this application) can create exercises consisting of pieces to be read by the students. The system will record and recognise the reading of the student and provide feedback on the correctness of the words read.

5.3. Tool for articulation therapy

Speech therapy helps improve communication ability and produces benefits in terms of quality of life and participation in society. It is however time-consuming, and patients rarely receive sufficient therapy to maximise their communication potential [11,12]. In articulation therapy speech therapists work with patients on the production of specific speech sounds and provide feedback on the quality of these speech sounds. Previous research shows that computer programs using speech recognition can improve outcomes of speech therapy for adults with speech difficulties [13].

We have developed a browser-based application that can be used to support articulation therapy. The application demonstrates another specific use of the CloudCAST platform that can be extended for other needs.

The application supports individual user practice of words that are set in sentences. The user or their therapist can set a series of exercises, consisting of the sentences they should practice. Visual feedback on the accuracy of the target sounds in the sentence is provided.

6. Discussion

CloudCAST aims to create a self-sustaining community of academic and speech professionals which will continue to grow after its 3 year funding period. It is our belief that only by collaborating in this way can we make the benefits of speech technology available to those who need it most and at the same time create the knowledge bases for

further technical improvement. To attain critical mass we need to widen the participants beyond the initial partners.

Therefore we are reaching out via publications such as this to encourage developers and practitioners to join our community. We warmly welcome any contributions and are happy to support any enquiries from interested developers and other professionals.

References

[1] A.N. Author, *Book Title*, Publisher Name, Publisher Location, 1995.

[2] A.N. Author, Article title, *Journal Title* **66** (1993), 856–890.

[3] Clapham, R., Middag, C., Hilgers, F., Martens, J.-P., van den Brekel, M., & van Son, R. (2014). Developing automatic articulation, phonation and accent assessment techniques for speakers treated for advanced head and neck cancer. *Speech Communication*, **59**, 44–54.

[4] Saz, O., Yin, S.-C., Lleida, E., Rose, R., Vaquero, C., & Rodríguez, W. R. (2009). Tools and Technologies for Computer-Aided Speech and Language Therapy. *Speech Communication*, **51**, 948–967.

[5] Christensen, H., Cunningham, S. P., Fox, C., Green, P., & Hain, T. (2012). A comparative study of adaptive, automatic recognition of disordered speech. In *InterSpeech* 2012.

[6] Yamagishi, J., Veaux, C., King, S., & Renals, S. (2012). Speech synthesis technologies for individuals with vocal disabilities : Voice banking and reconstruction. *Acoustic Science and Technology*, **33**, 1–5.

[7] Amazon.com Inc. (n.d.). Alexa Voice Service. Retrieved February 17, 2017, from http://developer.amazon.com/alexa-voice-service

[8] Google.com. (n.d.). Google Speech API. Retrieved February 17, 2017, from http://cloud.google.com/speech

[9] Mullin, E. (2016). Why Siri Won't Listen to Millions of People with Disabilities. Scientifc American.

[10] Povey, D. (2017). Kaldi. Retrieved February 17, 2017, from http://kaldi-asr.org

[11] Alumae, T. (2016). Kaldi GStreamer Server. Retrieved February 17, 2017, from https://github.com/alumae/kaldi-gstreamer-server

[12] Anonymous in the draft

[13] Law, J., Garrett, Z., & Nye, C. (2003). Speech and language therapy interventions for children with primary speech and language delay or disorder. Cochrane Database of Systematic Reviews, (5).

[14] Enderby, P., & Emerson, L. (1995). Does Speech and Language Therapy Work? London: Singular.

[15] Palmer, R., Enderby, P., & Hawley, M. (2007). Addressing the needs of speakers with longstanding dysarthria: computerized and traditional therapy compared. International Journal of Language & Communication Disorders / Royal College of Speech & Language Therapists, 42 Suppl 1, 61–79.

[16]