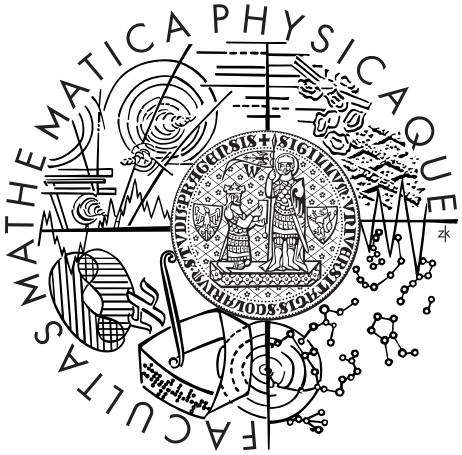
Charles University in Prague
Faculty of Mathematics and Physics

MASTER THESIS



Ondřej Klejch

Development of a cloud platform for automatic speech recognition

Institute of Formal and Applied Linguistics

Supervisor of the master thesis: Mgr. Ing. Filip Jurčíček Ph.D.
Study programme: Informatics
Specialization: Theoretical Computer Science

Prague 2015

Dedication.

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular the fact that the Charles University in Prague has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 paragraph 1 of the Copyright Act.

In ..... date ..... signature of the author

Název práce: Development of a cloud platform for automatic speech recognition

Autor: Ondřej Klejch

Katedra: Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: Mgr. Ing. Filip Jurčíček Ph.D., Ústav formální a aplikované lingvistiky

Abstrakt:

Klíčová slova:

Title: Development of a cloud platform for automatic speech recognition

Author: Ondřej Klejch

Department: Institute of Formal and Applied Linguistics

Supervisor: Mgr. Ing. Filip Jurčíček Ph.D., Institute of Formal and Applied Linguistics

Abstract:

Keywords:

Contents

Introduction

The most natural form of people's communication is speech. In order to be able to talk with a computer, it is crucial to have a good Automatic Speech Recognition (ASR) system. On one hand, there are several open-source ASR toolkits, however deployment of such toolkits requires substantial knowledge therefore for common software developers it's not easy to use them. On the other hand, there are a few webservices that provide ASR as a service, yet these webservices don't solve all problems - either they are paid, closed-source or they are not customizable. So **the first goal of the thesis is to develop a cloud platform for ASR** that is easy to use both from user's and maintainer's point of view.

Although accuracy of ASR systems is improving, these systems are still far from perfect. One of the reasons is that accuracy of ASR systems relies heavily on the amount of the training data and there isn't enough publicly available transcribed speech data. By providing free ASR webservice it is possible to collect vast amount of recordings that can be manually transcribed and used later on for further research. Consequently, **the second goal of the thesis is to create an annotation interface** so that recordings obtained by CloudASR platform can be annotated and given back to the community.

In the following text there will be described development and deployment of CloudASR platform and of its annotation interface. Chapter 1 introduces ... In Chapter 2 architecture of CloudASR is described. Annotation interface and theory related to obtaining of human transcriptions is presented in Chapter 3. Finally, Chapter 5 concludes this thesis. User manual and programmer manual can be found in the Attachments.

1. Title of the first chapter

1.1 Title of the first subchapter of the first chapter

1.2 Title of the second subchapter of the first chapter

2. Title of the second chapter

2.1 Title of the first subchapter of the second chapter

2.2 Title of the second subchapter of the second chapter

Conclusion

Goals of this thesis were to develop a cloud platform for ASR, CloudASR, and an annotation interface for annotating speech data. These goals were successfully accomplished and in several aspects even surpassed - in addition to original requirement to create batch recognition mode, we also implemented online recognition mode. In the following sections we summarize our achievements in detail and at the end we propose ideas for future work.

Cloud platform for ASR

The first goal of this thesis was to develop a cloud platform for ASR, CloudASR, that would provide batch API for speech recognition of wave files. The platform uses Master/Worker architecture. Consequently, it is able to run both on single-machine and multi-machine setup. The platform allows us to run workers for various language models and to scale workers according to our needs. To be able to run CloudASR on several machines we chose Mesos/Marathon as an underlying technology The current implementation of the API supports two modes of speech recognition: batch and online.

Firstly, batch mode allows users to send a file with a recording to the server and then it sends transcribed text back as a json. API of this mode is similiar to Google Speech API which allows users to switch from Google Speech API to CloudASR easily.

Secondly, users can transcribe speech recordings in real-time via online mode. We have also created Python and JavaScript libraries for using our API. JavaScript library achieves similiar latency as WebkitSpeechRecognition in Google Chrome !!add benchmark!!.

Finally, we wanted CloudASR to be easily deployable. Because of that, we used Docker for creating and running application containers. As a result only dependency that users have to install is Docker for single-node setup and Mesos Cluster for multi-node setup. Moreover, installation scripts for these dependencies are included within the distribution together with deployment scripts, that can be used for CloudASR instances management.

Annotation interface

The second goal of this thesis was to create an annotation interface for annotating speech data. First responsibility of the annotation interface is to collect and store obtained recordings.

The second responsibility is to allow users to rate transcriptions of the recordings (Is the transcription correct? yes/no) or to subsequently add their own transcriptions. The annotation interface implements algorithm to choose golden transcription from several manual transcriptions that were obtained for the recording. Additionally it is also possible to add manual transcriptions via external job at CrowdFlower.

The third responsibility is to provide export of transcribed recordings. This can be done either by downloading archive from the web or by using Torrent.

Future work

- Since manual transcription of recordings is expensive it would be good to make users transcribe only parts of the recordings in which ASR system wasn't confident enough !!cite (http://www.phontron.com/paper/sperber14slt.pdf)!!. This idea could be used for both user transcription and CrowdFlower transcription.

- With manually transcribed recordings from CloudASR platform it is possible to continuously improve accuracy of the underlying ASR system by adapting the language model to the type of language that the users of the CloudASR really use. Thus CloudASR could provide an option to automatically update language model when a certain amount of new transcribed recordings was collected.

- Because running CloudASR platform is expensive in terms of costs for a server hosting, it would be good to optimize usage of individual workers so that spare workers are shut down when there is no need for them and new workers are started when the traffic arise. This can be achieved either by providing feedback control based systems !!cite (http://shop.oreilly.com/product/0636920
or by using machine learning techniques. !!cite!!

- As CloudASR platform provides API for speech recognition, it could also be used for another speech related tasks like Language Identification, Speaker Identification, Voice Activity Detection, etc.

Bibliography

List of Tables

List of Abbreviations

Attachments