ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/307637720

A distributed platform for speech recognition research

Conference Paper · June 2016

CITATIONS
0

READS
12

2 authors:

Zhanibek Kozhirbayev
Nazarbayev University
6 PUBLICATIONS **21** CITATIONS
SEE PROFILE

Shynggys Islam
Nazarbayev University
3 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Data analytics in retail View project

All content following this page was uploaded by Zhanibek Kozhirbayev on 06 September 2016.

The user has requested enhancement of the downloaded file.

UDC 004.2

KOZHIRBAYEV ZH., ISLAM SH.

A DISTRIBUTED PLATFORM FOR SPEECH RECOGNITION RESEARCH

(Nazarbayev University, National Laboratory, Astana, Kazakhstan)

Abstract

Distributed and parallel processing of big data has been applied in various applications for the past few years. Moreover, huge advancements took place in usability, economic efficiency, and multiplicity of parallel processing systems, with big data analysis and speech recognition research supported by many researchers.

In this paper we examined and investigated which parts of speech recognition research may be parallelized and computed using distributed computing platforms. Firstly, we address the case of efficiently computing n-gram statistics on MapReduce platforms to build a language model (LM). Secondly, we show how the Automated Speech Recognition (ASR) tool can work efficiently regarding the speed and fault-tolerance in distributed environment such as Sun GridEngine (SGE).

Keywords: Distributed Computing, Sun GridEngine, Hadoop ecosystem, MapReduce

1. Introduction

The automatic speech recognition area went through several major progresses in the past decade initiated by changes in algorithms, signal processing, system architectures and hardware. The last two aspects play a significant role in Speech Recognition Research. The trigger for the development of distributed computing is the affordability of the cost effective, powerful machines as well as network tools. Several high-powered machines that are connected to one another make the total achievable computing power significantly broad. This kind of system might perform greater results rather than a single powerful machine. Distributed computing is the decentralized way of dealing with the computing stages of the application which can be distributed among the linked machines.

The remainder of this paper is organized as follows. Section 2 describes the parallelization technologies which might be applied in the speech recognition research. To be precise, the Hadoop ecosystem, which can be employed in building a language model when the size of the language corpus is big; and the Sun GridEngine, which distributes the tasks such as data alignment and audio decoding, will be presented in this section. Section 3 demonstrates the results obtained during the experiments. Finally, the last section concludes the paper and suggests further investigations in this area.

2. Parallelization technologies

During the research the parts of the speech recognition processes are examined in order to identify the tasks which may be parallelized. The two major tasks were distinguished which take a while when they are running on one single machine. Therefore, the distributed computing was applied to these processes and they will be described in this section in more details.

2.1 MapReduce in building LM

In this work, we address the problem of efficiently computing n-gram statistics on MapReduce platform. This is needed to build a language model which will be later converted to the ARPA format.

MapReduce [2, 3] is used widespread since the past few years as a programming model as well as its open-source realization Hadoop. A platform for parallel data computing is supplied by MapReduce. It enforces a harsh programming model; however, it provides its users with technical options such as dealing with machine errors as well as an automatic spread of the processing. In order to utilize it effectively, issues have to be cast into its programming model, considering its characteristic features.

In previous experiments only one single machine was used to build a language model. It takes a while to process such a big corpus to generate unigrams, bigrams and trigrams. Moreover, the corpus might be increased in the future and the necessity to build a language model will rise again. Therefore, the Hadoop cluster was built on seven nodes where each node has 8 Gb RAM and 4 cores. The results that obtained using MapReduce platform were significant which can be seen in Section 3.

Table 1

Comparison of the computing environments

Single machine		Hadoop cluster (7 nodes)	
# of cores	RAM	# of cores	RAM
4	16 Gb	28	56 Gb

2.2 Parallelization in Kaldi

A toolkit named Kaldi [1] was used for speech recognition research. The perfect condition for processing is a cluster of Linux nodes using SGE, with the admission to shared folders through either NFS or similar network filesystem [4, 5]. The perfect form of processing environment as well as required limits to perform Kaldi will be explained below in this section.

The speech recognition research using Kaldi toolkit was conducted in one single machine since it can be easily configured to run on a single machine if it is a supercomputer. However, the machine used in the research has 16 cores and only 32 Gb RAM. Kaldi toolkit performs some tasks sequentially and some tasks parallel. For example, the data alignment and audio decoding jobs are run parallel. Also, the aspect which should be considered during the research is the size of the language model. The decoding task depends on the size of the LM and requires approximately 6 Gb RAM. Therefore, only 5 cores of the single machine are useful for the recognition process. This approach was inefficient. Therefore, the new approach using SGE was conducted because of the availability of the cost effective, much powerful nodes and network tools.

Sun GridEngine is the open-source grid control instrument which is used widespread. Recently Oracle started supporting SGE and renamed it Oracle GridEngine [4]. The currently used version in the mentioned system is 6.2u5; SGE is time proven and earlier made versions are still stable and widely used. Furthermore, different open-source possible tools to SGE do exist, however, built platform in the mentioned system refer to the version that is nowadays supported by Oracle.

Table 2

Comparison of the computing environments

Single machine		SGE cluster (17 nodes)	
Total	Used for recognition	Total	Used for recognition
16 cores	5 cores	144 cores	46 cores
32 Gb RAM	30 Gb RAM	336 Gb RAM	276 Gb RAM

The grid cluster was build using 13 machines where each node has 8 cores and 16 Gb RAM, and 4 virtual machines, that are deployed on Openstack, with 8 cores and 32 Gb RAM. The results which obtained using SGE platform were significant which can be seen in Section 3.

3. Experiments and Results

This section provides the results of both building the language model and audio decoding. It can be seen from the below tables that results given by Hadoop and SGE significantly overcomes results by single machine.

Table 3

Results of building LM

# of runs	Corpus and its size	NGrams	Elapsed time	
			Single machine	Hadoop cluster
1	Kazcorpus 1.6G	Unigram	>> 4 hours	19mins, 55sec
2		Bigram	>> 4 hours	20mins, 28sec
3		Trigram	>> 4 hours	20mins, 19sec

Table 4

Results of audio decoding

# of runs	Data Set	Elapsed time	
		Single machine	SGE cluster
1	10 hours audio set	≈ 3184mins	≈ 385mins

4. Conclusion

In this paper, we have applied MapReduce to compute n-gram statistics for the language model. Moreover, the ideal environment which has a SGE installed in it may provide a significant improvement for Kaldi toolkit. Both of these applied platforms decrease the processing time in a sufficiently great way. Further investigations will be conducted to explore new features of distributed computing.

To sum up, we will continue to improve the Speech Recognition Research in terms of parallelization.

Acknowledgments

The authors would like to thank the National Laboratory Astana for the resources used to perform these investigations and Karabalayeva M. and Yessenbayev Zh. for providing results of audio decoding that added a value in better analysis for this paper.

References:

1. Povey, D, Ghoshal, A, Boulianne, G, Burget, L, Glembek, O, Goel, N, Hannemann, M, Motlicek, P, Qian, Y, Schwarz, P, Silovsky, J, Stemmer, G & Vesely, K 2011, "The Kaldi Speech Recognition Toolkit", IEEE 2011 Workshop on Automatic Speech Recognition and Understanding

2. Zaharia, M 2014, Introduction to MapReduce and Hadoop, UC Berkeley RAD Lab

3. MapReduce, viewed 15 April 2016, URL https://hadoop.apache.org

4. Sun microsystems 2009, Sun N1 Grid Engine 6.1 User's Guide, Santa Clara, CA, USA

5. Open Grid Engine, viewed 15 April 2016, URL http://gridscheduler.sourceforge.net