A GPU-Based Cloud Speech Recognition Server For Dialog Applications

Alexei V. Ivanov,
Verbumware Inc.

NVIDIA GTC,
San Jose, April 5, 2016

GPU-based Baseline System Inference Statistics

TASKS\LMs	BCB05ONP	BCB05CNP	BCB05ONP	BCB05CNP	TCB20ONP	BCB05ONP	BCB05CNP	TCB20ONP
NOV'92 (5K) WER	**5.66%**	2.30%	**5.66%**	2.30%	1.85%	5.77%	2.19%	1.63%
NOV'92 (5K) 1/xRT	2.15	2.14	**30.58**	**30.49**	**27.47**	5.08	5.26	4.54
NOV'93 WER	18.22%	**19.99%**	18.22%	**19.99%**	7.77%	18.13%	20.19%	7.63%
NOV'93 1/xRT	2.15	2.15	**30.12**	**30.21**	**26.67**	4.33	4.20	3.90
Power/RTchan.	**~3.6W**		**~9 W**			from 75**W** (1 ch) to **15W** (full load)		
Hardware	**Tegra K1 (32 bit)**		**GeForce GTX TITAN BLACK**			i7-4930K @3.40GHz		
	GPU-enabled					Nnet-latgen-faster		

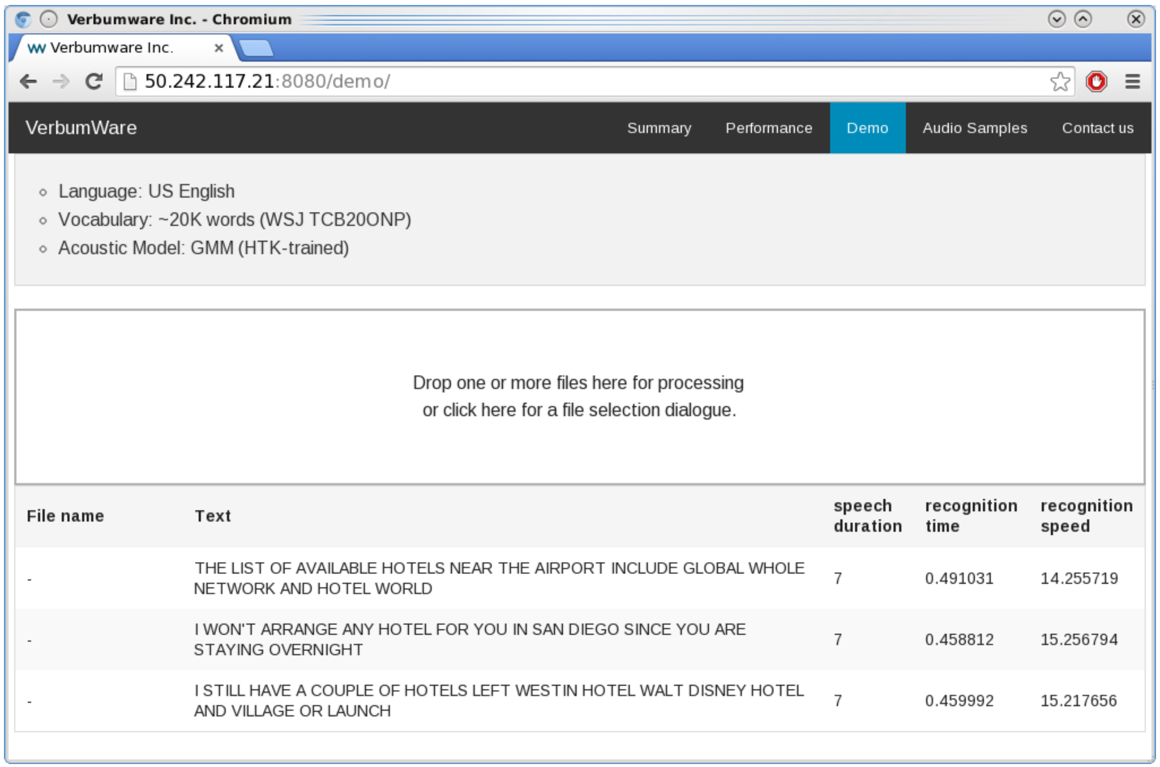
- **Accuracy** of our GPU-enabled engine **is approximately equal** to that of the reference implementation. There is a small fluctuation of the actual WER (mainly) due to the differences in arithmetic implementation.

- For the single-channel recognition **the TITAN-enabled engine is significantly faster** than the reference. This is important in tasks like media-mining for specific a priori unknown events.

- Our implementation of the speech recognition in the **mobile** device (Tegra K1) enables **twice faster than real-time processing** without any degradation of accuracy.

- Our GPU-enabled engine allows **unprecedented energy efficiency** of speech recognition. The value of 15W per RT channel for i7-4930K was estimated while the CPU was fully loaded with 12 concurrent recognition jobs. This configuration is the most power efficient manner of CPU utilization.

ASR Demo WEB Interface



ALTERNATIVES:

Google Speech API

Microsoft Prj Oxford

Amazon Alexa

IBM Watson

Nuance

COST $0.02-0.05/min

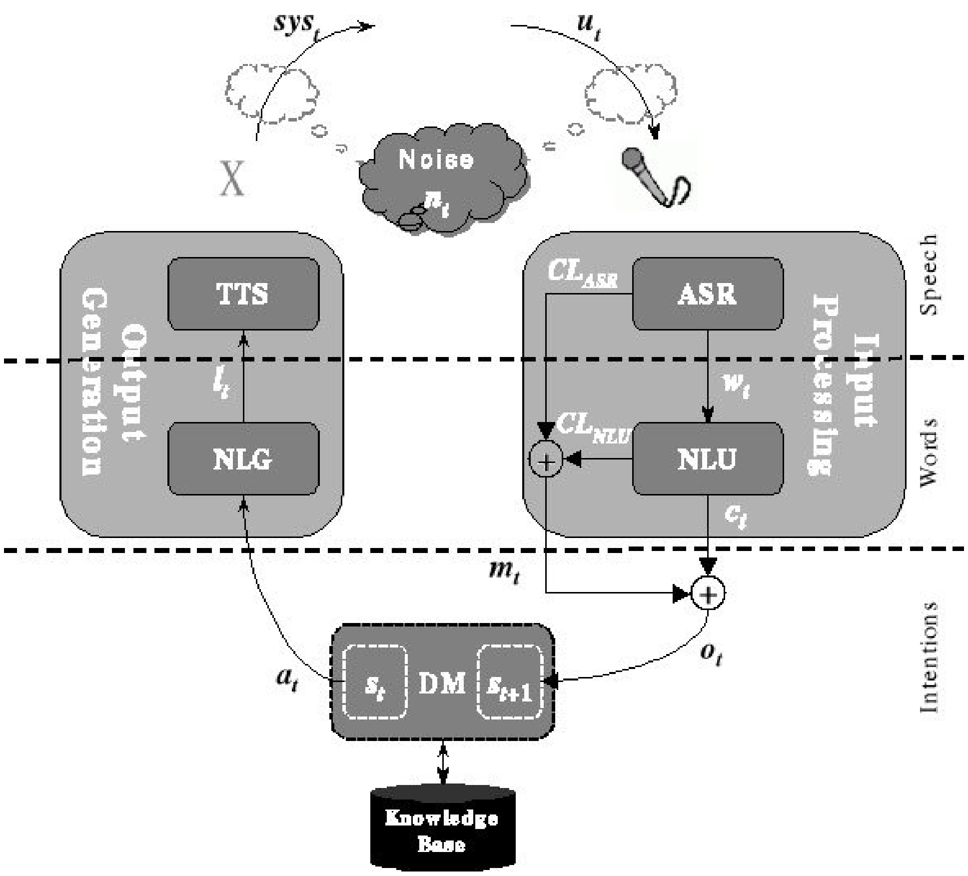
1 month to pay for DGX-1

http://verbumware.org:8080/demo

Browser-based Microphone Demo is coming soon

Speech Recognition in Dialogue Systems

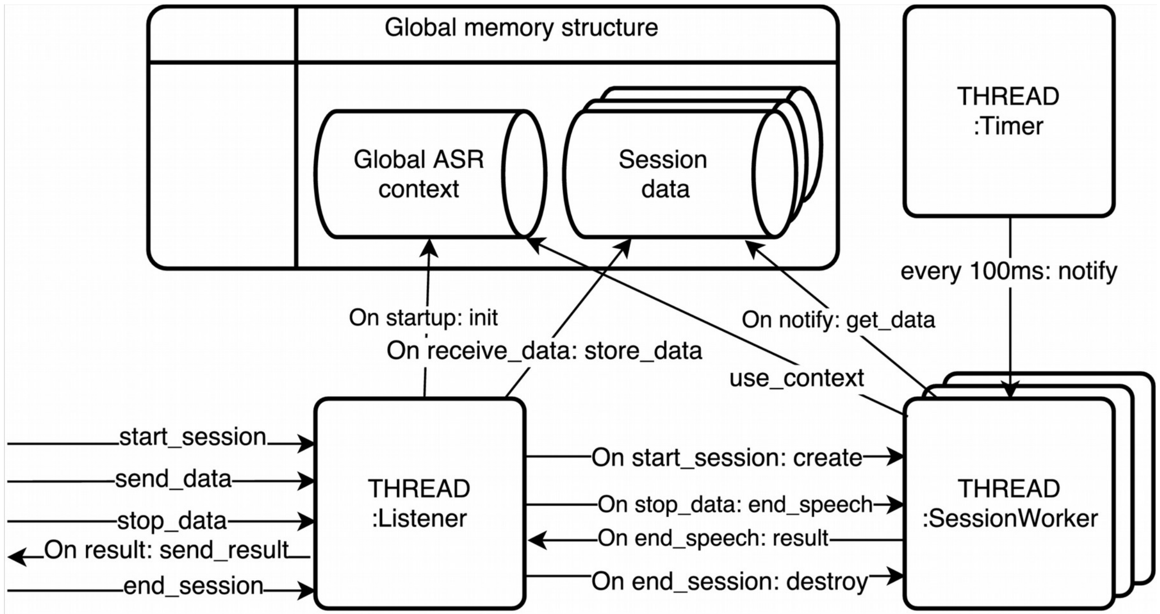
- **SDS cycle** User Input - User Output:
  - Recognition (ASR) & Understanding (NLU)
  - Dialog Management (DM)
  - Language Generation (NLG + TTS)
- **Difficulties**:
  - Time limits of the **natural communication**
  - **Spontaneous** speech: (Agramatism, Colloquialism, Back-channel, etc.)
  - **Speaker** properties **variation**



What needs to be changed?

- **Online processing** - start processing before recording is finished
- **Partial result** - report current best before the end of the utterance end
- **Partial back-tracking** - determine the part of the current partial best that is not going to be changed
- **Rapid model adaptation** - change model parameters to optimally suite the current speaker
- **Chunked processing** – less possibilities to exploit data-parallelism, no random access to the content

ASR System Architecture



Multi-threaded server wrapper architecture, memory object sharing within the single process

Online processing, **incremental output synthesis/presentation**

WEB-enabled (full-duplex asynchronous web-socket interface)

GPU processing is **cycling over processing** stages **in the job pool**

(! EACH CLIENT SPEAKS NO FASTER THAN THE NATURAL PACE !)

GPU Processing Schedule

Q: What is the optimal chunk size from the computational efficiency perspective?

A: Processing in **chunks is more preferable** as it reduces the required memory bandwidth (models are much larger than the data). Empirical estimate of a **sufficiently large chunk ~ 50 frames (0.5 sec)**, which poses a problem for interactive voice systems.

Q: What is the minimal specific latency the ASR server can have?

A: If we process in a **frame-synchronous manner (1 frame chunk)**, than the total ASR latency can be reduced **down to 150 ms** that is deemed acceptable for natural conversations.

OUR ASR SERVER IMPLEMENTS THE FRAME-SYNCHRONOUS (LOWEST LATENCY) PROCESSING

Statistical Modeling & Experimental Evaluation

Training:

Audio @ 8 kHz

760 hours of the target domain manually transcribed speech

AM: p-norm DNN with 4 hidden layers

LM: estimated on 5,8 million tokens; 525K tri-grams and 605K bi-grams over a lexicon of 23K words.

The decoding graph was compiled having approximately

5,5 million states and 14 million arcs.

Evaluation:

DEV contains 593 utterances (~ 10 h)

(68329 tokens, 3575 singletons, 0% OOV rate)

TST contains 599 utterances (~ 10 h)

(68112 tokens, 3709 singletons, 0.18% OOV rate).

Speed–Accuracy Trade-off

Set	Adaptation	Prunning Beam	Hypotheses Number	WER	Npass
DEV	fMLLR	various	various	22.27%	3
DEV	Online	50	40 K	21.58%	1 (slow)
DEV	Online	11	7 K	21.95%	1
TST	Online	11	7 K	23.05%	1

CPU 1/xRT	CPU N_{RT}	Pow/RTchan	GPU 1/xRT	GPU N_{RT}	Pow/RTchan
~ 1.07	~2	~ 150 W	~4.12	~26	~ 10-15 W

The SDS needs to respond in a timely manner, **no multiple-pass recognition** is allowed

A system with online adaptation is capable of that at the cost of a **slight WER increase**

Fast GPU-based Online Decoding **(~ 32 times faster than speech pace)**
With LibriSpeech 200K words & tgsmall **~ 26 times faster**

Human Performance Comparison

With the TST set WER of **about 23,05% our proposed system** has reached the level of broadly defined average human accuracy in the task of non-native speech transcription.

Experts have average WER **around 15%**

While **crowd-sourcing workers** perform significantly worse at **around 30% WER**

Specific Application Requirements

ETS Mission: “**To advance quality and equity in education by providing fair and valid assessments, research and related services.**”

reliability

Does the assessment produce similar results under consistent conditions?

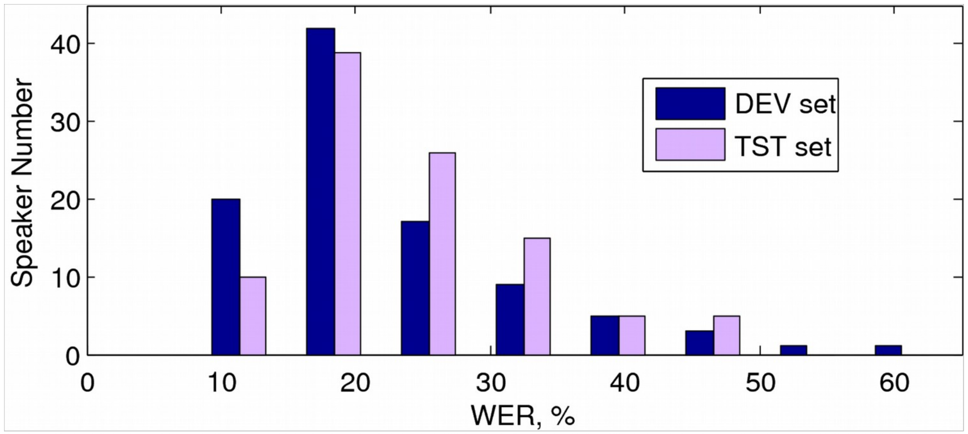
validity

Does the assessment measure what it is supposed to measure?

fairness

Does the assessment produce valid results for all subgroups of test takers?

Error Distribution over Speakers



Region	Speakers	p-value
Africa	10	0.84
South-East Asia	27	0.78
India	17	0.78
Americas	20	0.74
Europe and Central Asia	36	0.56
Middle East	28	0.31
Korea	30	0.13
China	27	0.02

ASR **accuracy** has to be studied as a **distribution** estimated on a broad target speaker population

There exists a **systematic limiting factor** precluding our ASR from sometimes showing low WERs (figure)

For the system to be **fair**, a stratification over any of the social groupings, (race, gender, geographical location, native language) shall not lead to a statistically significant alternation of the distribution (table)

We've developed a non-parametric method to evaluate error distribution miss-match

Online Model Adaptation

Identity-vector (i-vector) based

I-vector is continuously re-evaluated & fed to the DNN AM alongside the feature vector

I-vector computation involves:

- **evaluation of the GMM**

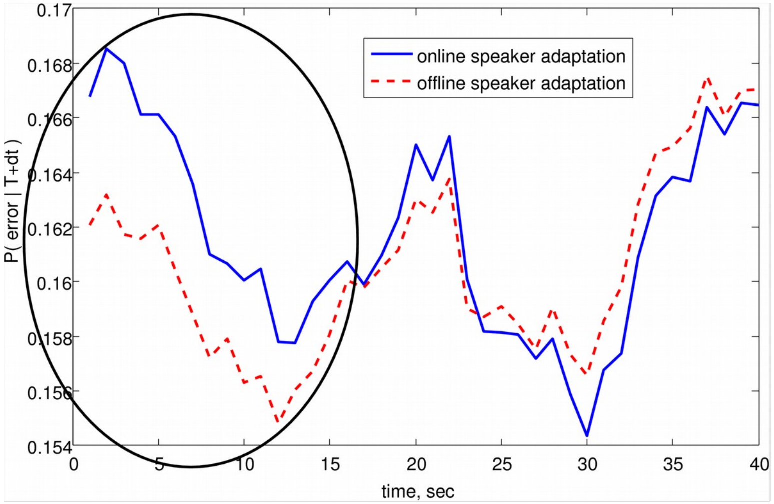
- a number of **vector operations** (e.g. normalization, etc.)

(100 times/sec)

- iterative **conjugate gradient descent** solution search

(~15 iterations @ 20 - 100 times/sec)

Error Distribution Over Time



Set	DEV	DEV
Adaptation	**Online**	Offline
Word Error Rate,%	**21.90**	21.74
Substitutions, %	**12.41**	12.25
Deletions, %	**5.93**	5.96
Insertions, %	**3.56**	3.54

The **online system** has higher WER in general (table) and particularly in **the beginning** of the utterance (figure)

Maintain the speaker adaptation **profile** through the whole dialog interaction

Initial interactions must be **simple** with a possibility of the **correct machine answer** regardless of the human input

Rhetoric structure in the figure ?

Error Distribution Over Word Type

TST Set	Total Words	Content Words	Function Words	Fillers+Interject.
Reference token count	67864	24596	37522	5746
Insertion count	2836	575	1649	612
Insertions, %	4.18%	**0.85%**	2.43%	0.90%
Mis-recocgnition count	12809	6740	5357	712
Mis-recocgnitions, %	18.87%	**9.93%**	7.89%	1.05%

Importance of an individual recognition error towards the general understanding of the interlocutor's input is not constant

(**23K content vs 319 function words + 24 interjections**)

Being an extremely small lexical set, **function words are more frequent** than content words in natural language

Some of the function word errors can be recovered by applying a content-conditioned **re-scoring model** that encapsulates **grammatical rules**

Content words follow the **minimal word constraint** -> less insertions

Recognition Result Post-Processing

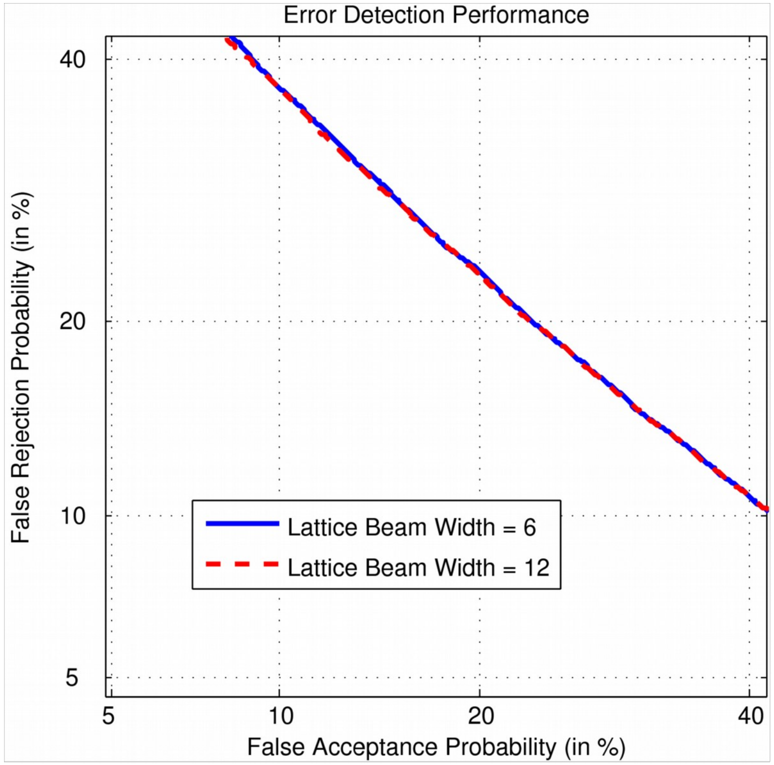
Analysis of the lattices & confusion networks allows to detect & recover recognition errors:

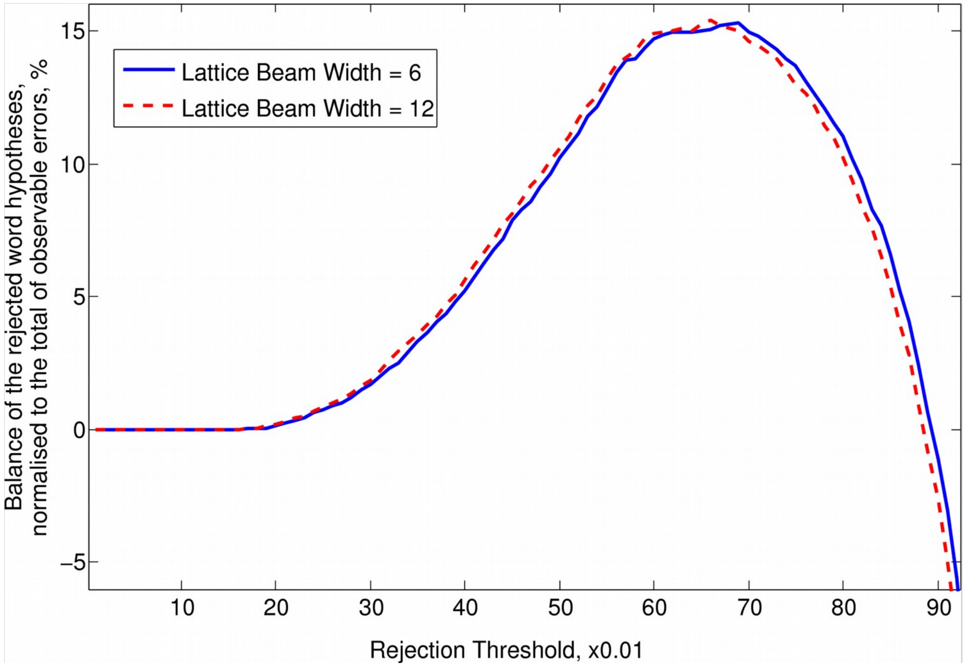
Essential when dealing with spontaneous speech

Practical if only it takes little time

Useful in the dialogue context as there is a possibility to recover via a number of dialogue strategies (e.g. clarification, confirmation, reprompt)

Error Detection





Confidence measure in our system = **posterior probabilities of word alternatives in the confusion network**

On the TST set the system **rejects 44,11% of true errors** and 6,38% of correct recognitions.

Increased complexity of confidence estimation **does not significantly alter** its performance

Conclusions

Building a **fast and accurate** dialog speech recognition **system** for interacting with distant non-native interlocutors **is possible**

The DNN with i-vector-based speaker adaptation = the **state-of-the-art** acoustic decoding **accuracy** with **single-pass processing** (not efficient in first 15 sec.)

Word **posterior probs in confusion networks** have power towards predicting errors. They can correctly **predict 44,11% observable recognition errors** at the cost of **falsely rejecting 6.38%** of correct recognitions

Error distribution across auto-semantic and function words roughly estimates the **upper bound of the improvement** in WER that can be achieved with the grammatical re-scoring model

A better job needs to be done in the training to **ensure fairness** of the resulting system

Q & A

Do you have questions?

www.verbumware.net info@verbumware.net

alexei_v_ivanov@ieee.org