# Conversational AI at Large Scale

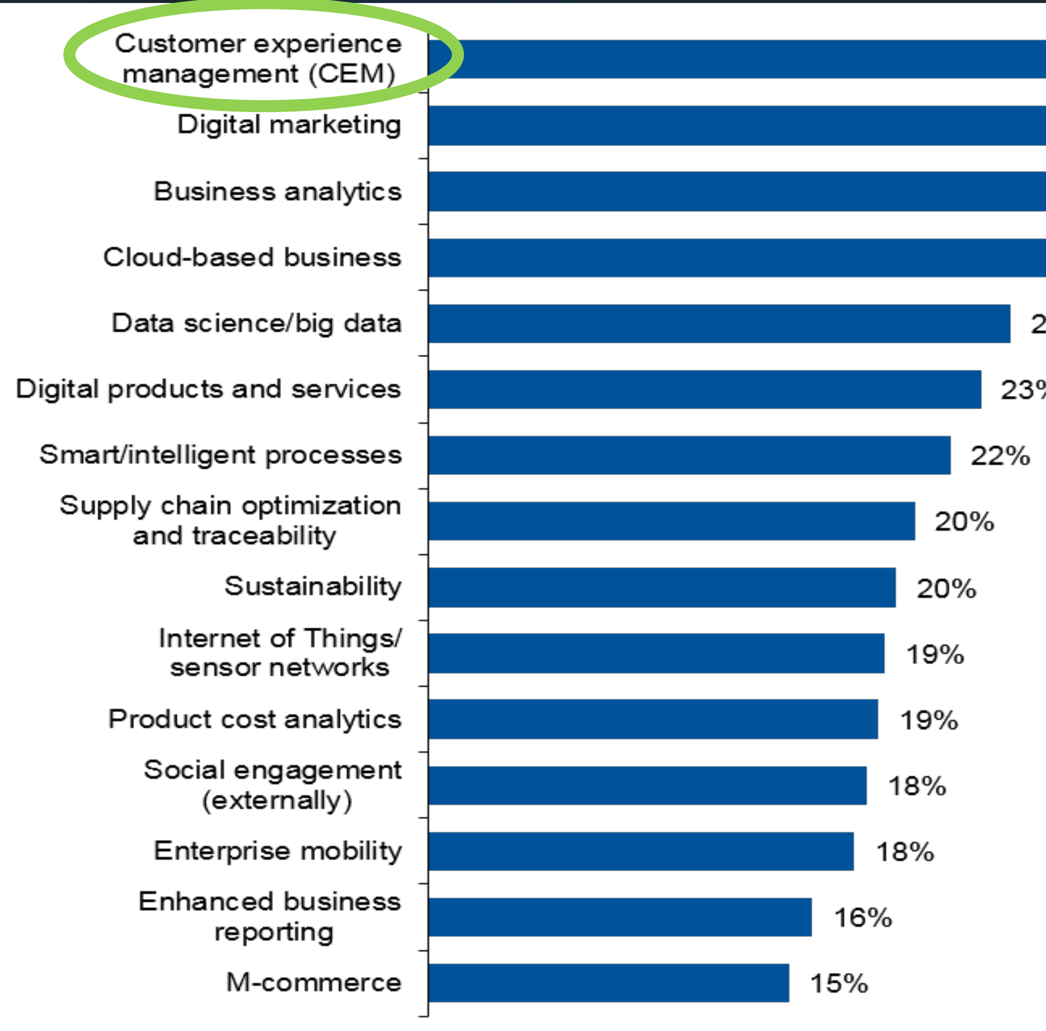
Yishay Carmiel

Spoken

# The Power of Conversations

# Customer Experience is the New Competitive Battlefield

CEOs' Five-Year Investment Intention Toward a Range of Modern Technology-Enabled Capabilities




**Less Than 15%**

| | |
|---|---|
| "Thinking"/intelligent computers | 14% |
| Social collaboration (internally) | 13% |
| Design thinking | 12% |
| Pay as you go | 10% |
| Mass customization | 10% |
| Location intelligence | 10% |
| Robot staffing | 9% |
| Gamification | 7% |
| Additive manufacturing | 7% |
| Neurobusiness | 6% |
| E-discovery | 6% |

## 88%

of organizations surveyed plan to increase customer experience technology investment

-Gartner

## 89%

of marketing leaders expect to compete primarily on the basis of customer experience as compared with 36% four years ago.

-Gartner

# Customer Experience in the Contact Center

In contact centers today…

**22** million agents

**>75%** of the interactions are still voice

**100** hours/month of talk time

**19.8B** hours of conversation/year

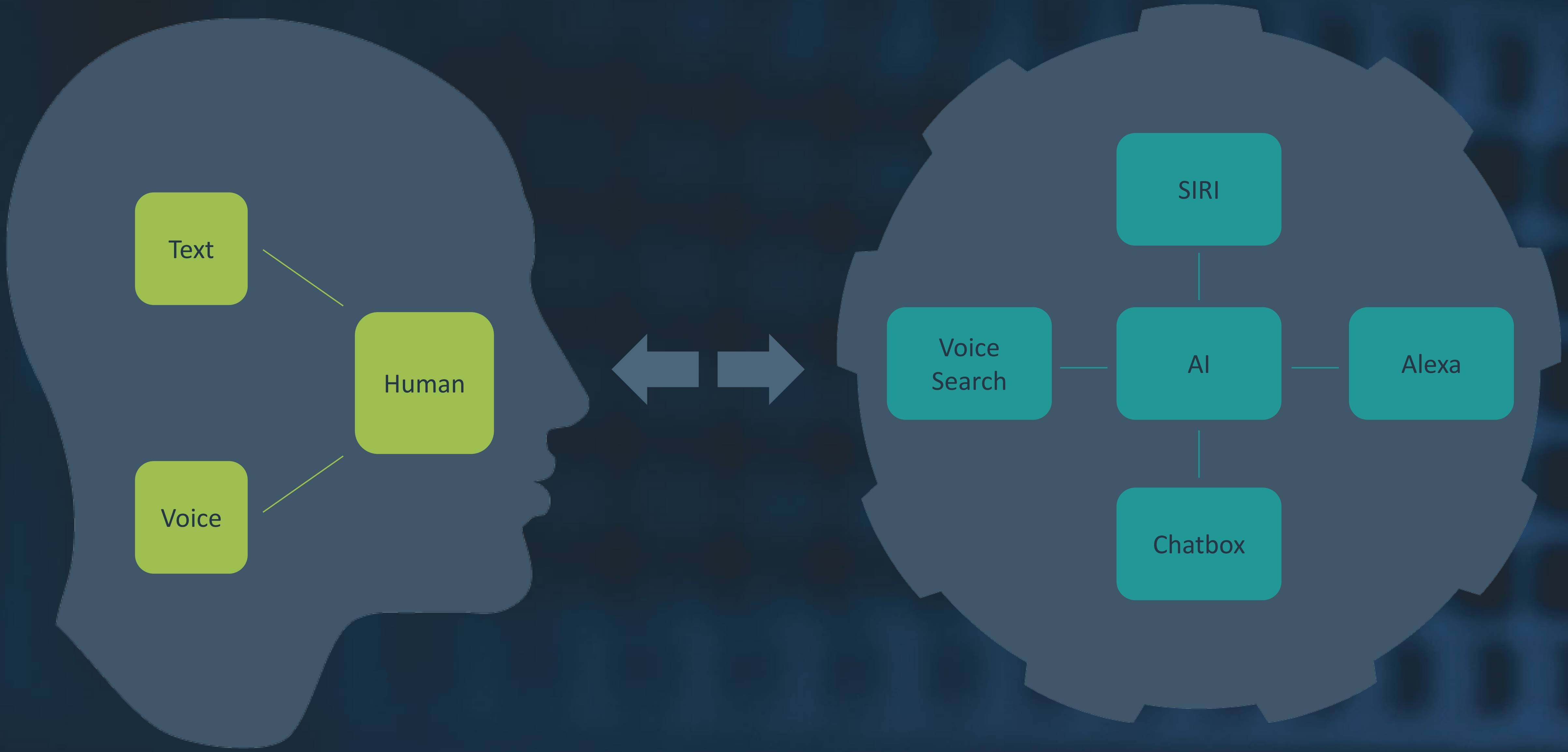
# Customer Experience in the Contact Center

And yet, the capabilities for optimizing customer experience on the voice channel are…

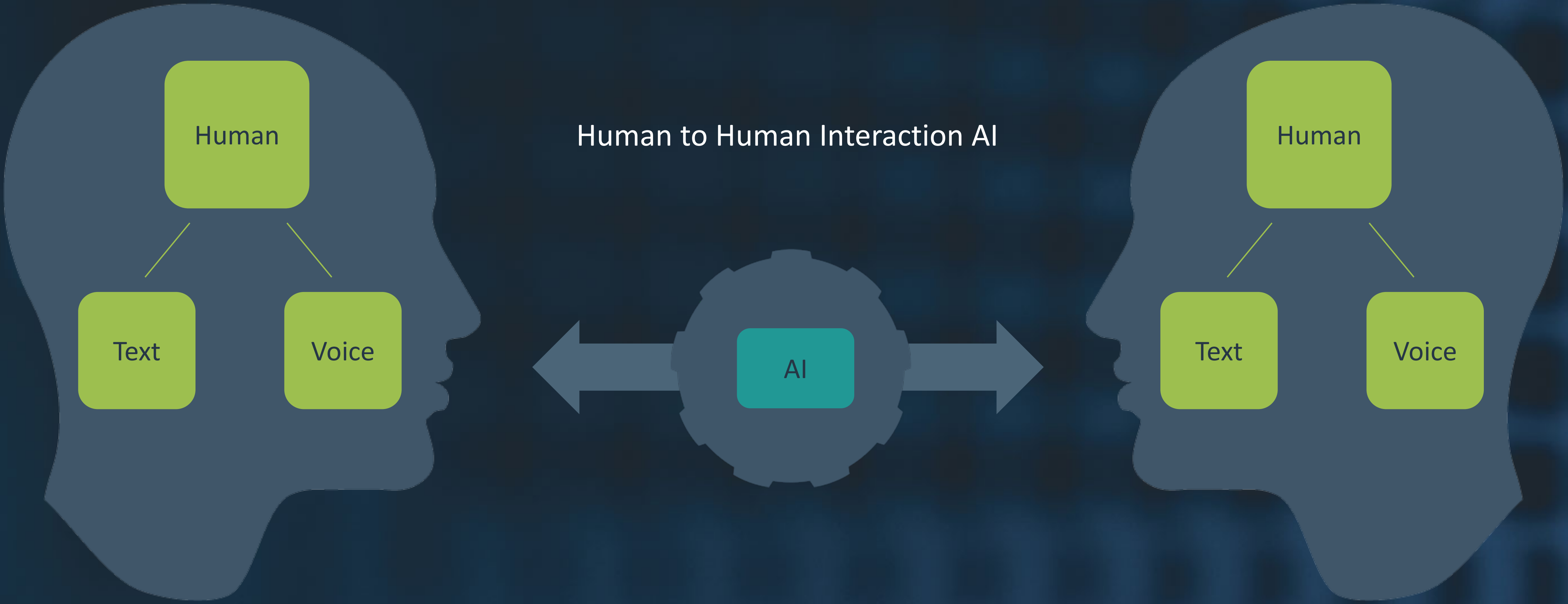
- Inadequate
- Imprecise
- Unresponsive
- Poorly integrated
- Bureaucratic
- Complicated
- Dumb

# Conversational AI

# Prevalent View | Man to Machine AI



# Spoken Conversational AI



# Passive vs. Active System

## Passive

- Offline analysis
- The system does not intervene during the conversation
- Uses closed conversations as input
- Can work in batch mode, allows a wider range of algorithms
- Great for identifying trends

VS

## Active

- Online analysis
- The system provides insights and recommendations to participants of the conversation or even takes action in real-time
- Uses an ongoing conversation as input
- Online, real-time algorithms

# Macro vs. Micro System

## Micro

- Deals with a single interaction (one phone call)
- Often the algorithms must be more accurate because their output is directly interpreted (i.e. in call summarization)

VS

## Macro

- Deals with a set of interactions (a million phone calls)
- Aggregates facts extracted from multiple interactions into global insights
- Leverages the rule of big numbers to go beyond imperfect results in isolated cases. Algorithms are designed for large datasets

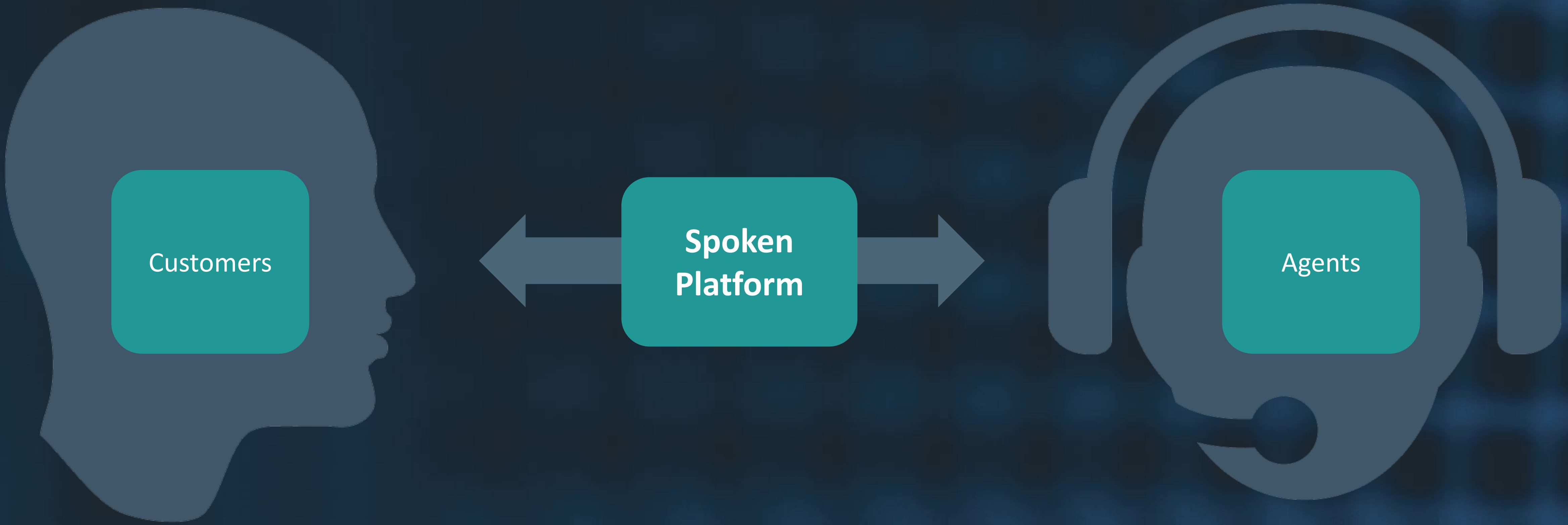
# How do we classify the use case?

The use case examples matrix

| | Passive | Active |
|---|---|---|
| **Micro** | 1. Conversation summarization<br>2. Meta-data extraction<br>3. Automatic note-taking | Smart AI assistants with dynamic recommendations from knowledge bases. |
| **Macro** | Sentiment analysis for all customers from NYC | 1. Identifying trends<br>2. Causes of negative sentiment<br>3. Outlier detection |

# Conversational AI at Spoken

# Spoken Conversation AI



# A few use cases

# The challenges

## 1,000,000

Analyzing 1,000,000h/day

## Fast & Accurate

Speaker Verification system

# Analyzing 1,000,000 hours/day

# Speech Recognition

# Speech Recognition System

## Signal Level

- Extracting only speech segments
- Feature Extraction
- User/Environment adaptation
- ***Deep Learning*** to generate features

## Acoustic Level

- Classification of the different sounds
- Adding contextual information on top of the sounds
- Using ***Deep Learning*** to do acoustic classifications

## Language Level

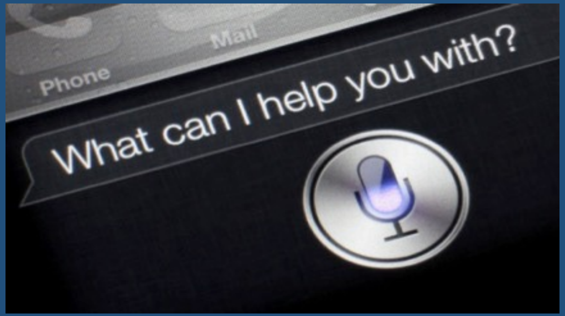
- Combining sounds into words
- Combining words into sentences
- Using ***Deep Learning*** to generate the different models

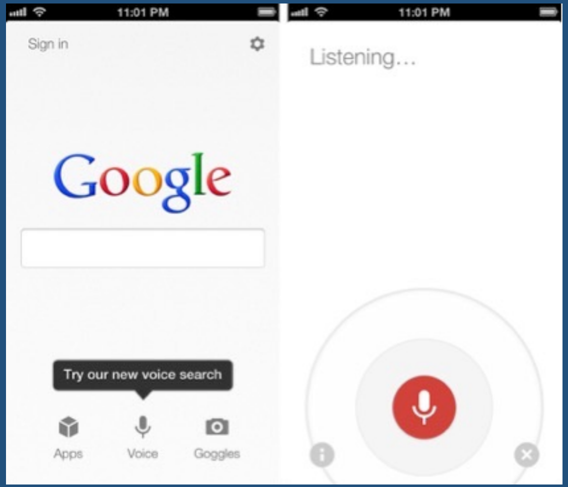
# Impact of Deep Learning on Speech Recognition

| Year | SWBD ERR | Relative Improvement | Overall Improved |
|---|---|---|---|
| 2008 | 23.6 | | |
| 2009 | 23.6 | | |
| 2010 | 23.6 | | |
| 2011 | 18.7 | 20.76271186 | |
| 2012 | 16.1 | 13.90374332 | |
| 2013 | 13.4 | 16.77018634 | |
| 2014 | 10.7 | 20.14925373 | |
| 2015 | 8 | 25.23364486 | |
| 2016 | 5.9 | 26.25 | 75 |
| *2017 | 5.5 | 6.779661017 | 76.69491525 |

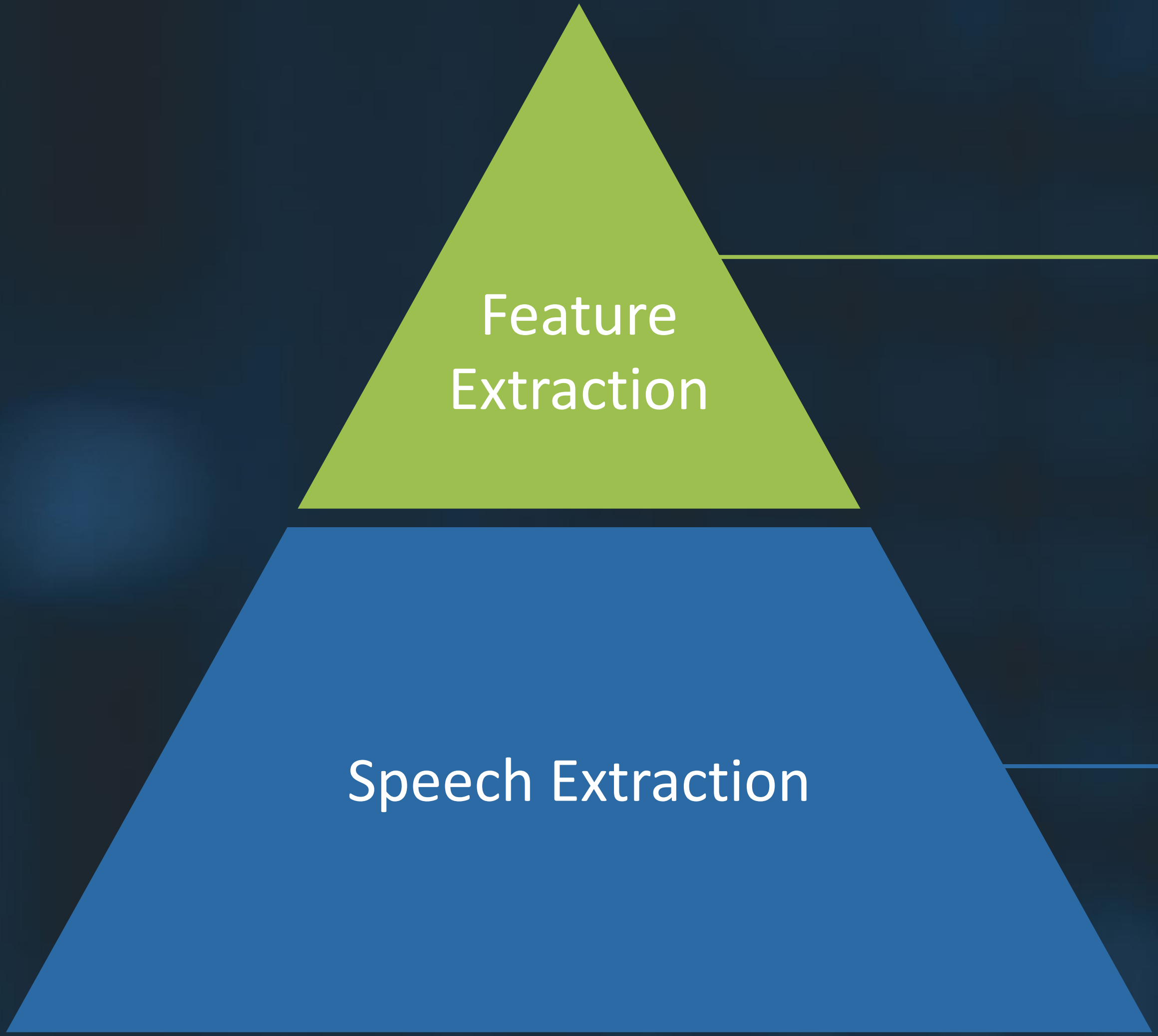

https://arxiv.org/abs/1703.02136

# Speech Recognition is Starting to Work

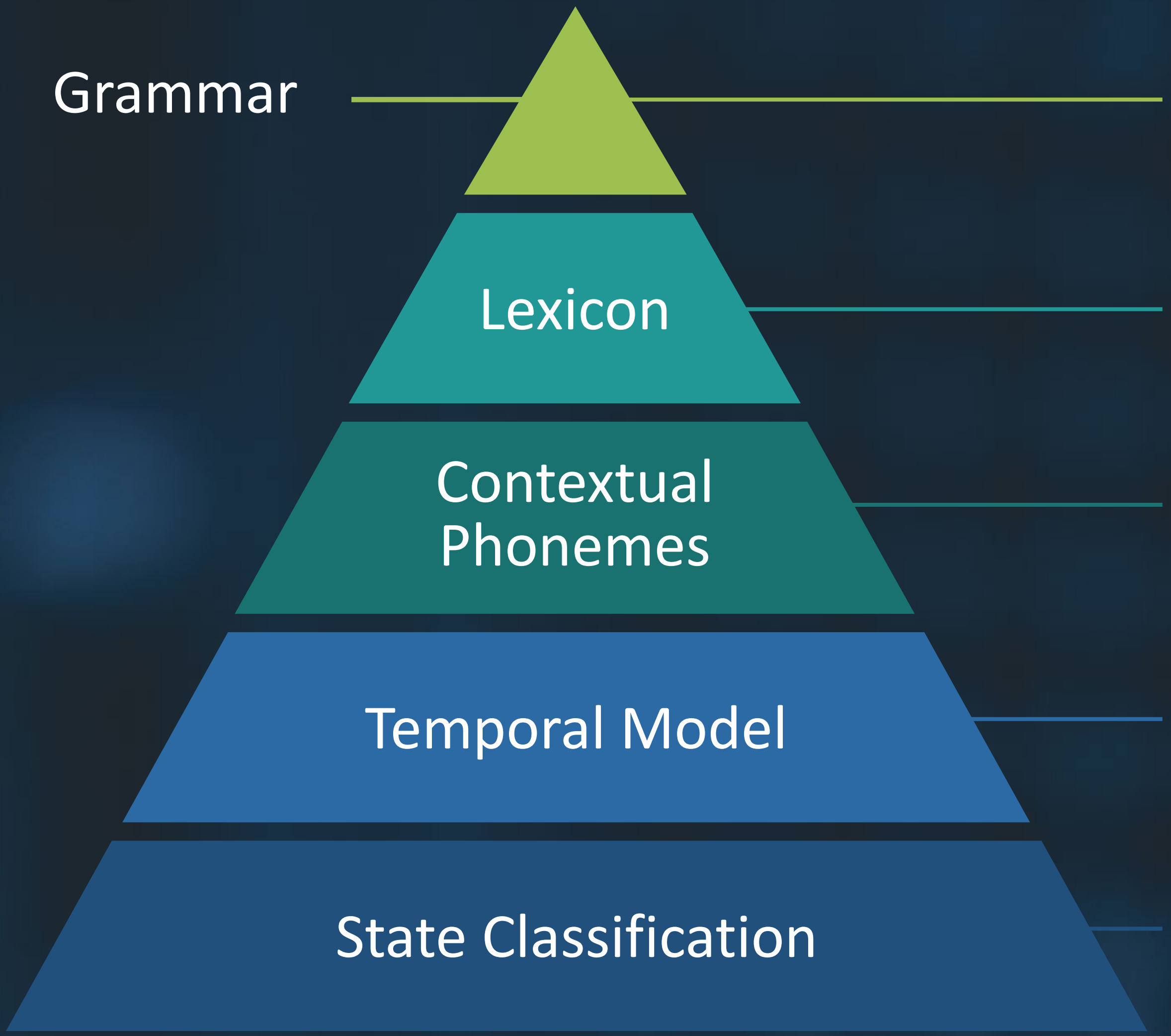
# Signal Level Analysis – Recent Advances




Extract features' parameters from the signal, Bottleneck Features using **DNN** or **CNN** as a baseline layer

Using **DNN** based **Voice Activity Detector** to extract only speech segments. Using **LSTM** for signal enhancement and Beamforming.

# Acoustic and Language Level Analysis – Recent Advances




**RNN** or **LSTM** techniques for language modeling

The vocabulary and their phonetic decompositions used to be generated manually or using data driven techniques. Recently we use **RNNs** and **LSTMs**

**Probabilistic model of contextual structure of phonemes**

**Modified HMM** or **RNN** classification. **CTC and frame subsampling models**

State classification is generated using **DNN** or a recursive network **(TDNN, RNN, LSTM)**

# Can we scale to 1,000,000 hours/day

# Is 1,000,000h/day A Realistic Number?

- Yes!
- Only in the contact centers there are millions of representatives
- 500,000h/day means analyzing ~60,000 representatives' conversations a day
- Actually 500,000h of conversations is bigger than 1,000,000h of speech (assuming that at least 2 people are interacting)

# Is 1,000,000h/day A Big Number?

- Yes!
- 1h of speech in standard quality is almost 60MB
- 1,000,000h of speech is 60TB a day
- This means applying state of the art deep learning models to 18PB/Year!

# 1,000,000 hours/day in $$$

State of the art.

### Real Time

- Lets assume the agents take calls for 10h every working day.
- This means processing 100,000h per hour.
- 100,000 real time conversations

### Algorithmic

- We need state of the art performance
- BLSTM network  is good but slow - x0.25 real time on a single core
- A slightly worse TDNN 25M parameters achieves 0.5 real time performance on a single core

### Scalability

- Assuming we use the TDNN model
- Need > 200,000 cores!
- >$40M in HW cost
- > $20M/year AWS prices

# What Can We do?

Three key points for optimization and acceleration

| | |
|---|---|
| **1. Algorithm** | 1. Frame Subsampling : New methods for reducing the search space.<br>2. Network Optimization: Parameters reduction and different topologies. Both for the acoustic model and language model |
| **2. Reducing Data Analysis** | Better Speech Extraction models – DNN Methods<br>Reducing Search Space by optimizing LM and lexicon |
| **3. HPC Methods** | Acceleration using GPU's, various optimization techniques from the HPC space. |

# Frame Sub-sampling

**“Purely sequence-trained neural networks for ASR based on lattice-free MMI” D. Povey et al**

## Original ASR System

| 10ms | 10ms | 10ms |
| --- | --- | --- |
| Features | Acoustic Score | FST Search |

## Frame Sub – Sampled

| 10ms | 30ms | 30ms |
| --- | --- | --- |
| Features | Acoustic Score | FST Search |

Result of acceleration by a factor of x3 – x9

# Speech Extracted Algorithm

## “MUSAN: A Music, Speech, and Noise Corpus” D. Snyder et al.

VAD (Voice Activity Detection) requires less CPU then Speech Recognition

We use machine learning to classify each frame – Noise, Speech, Music, Silence. Classifier can be GMM or DNN

Algorithms are either time domain or frequency domain based. The advanced ones use statistical signal processing techniques

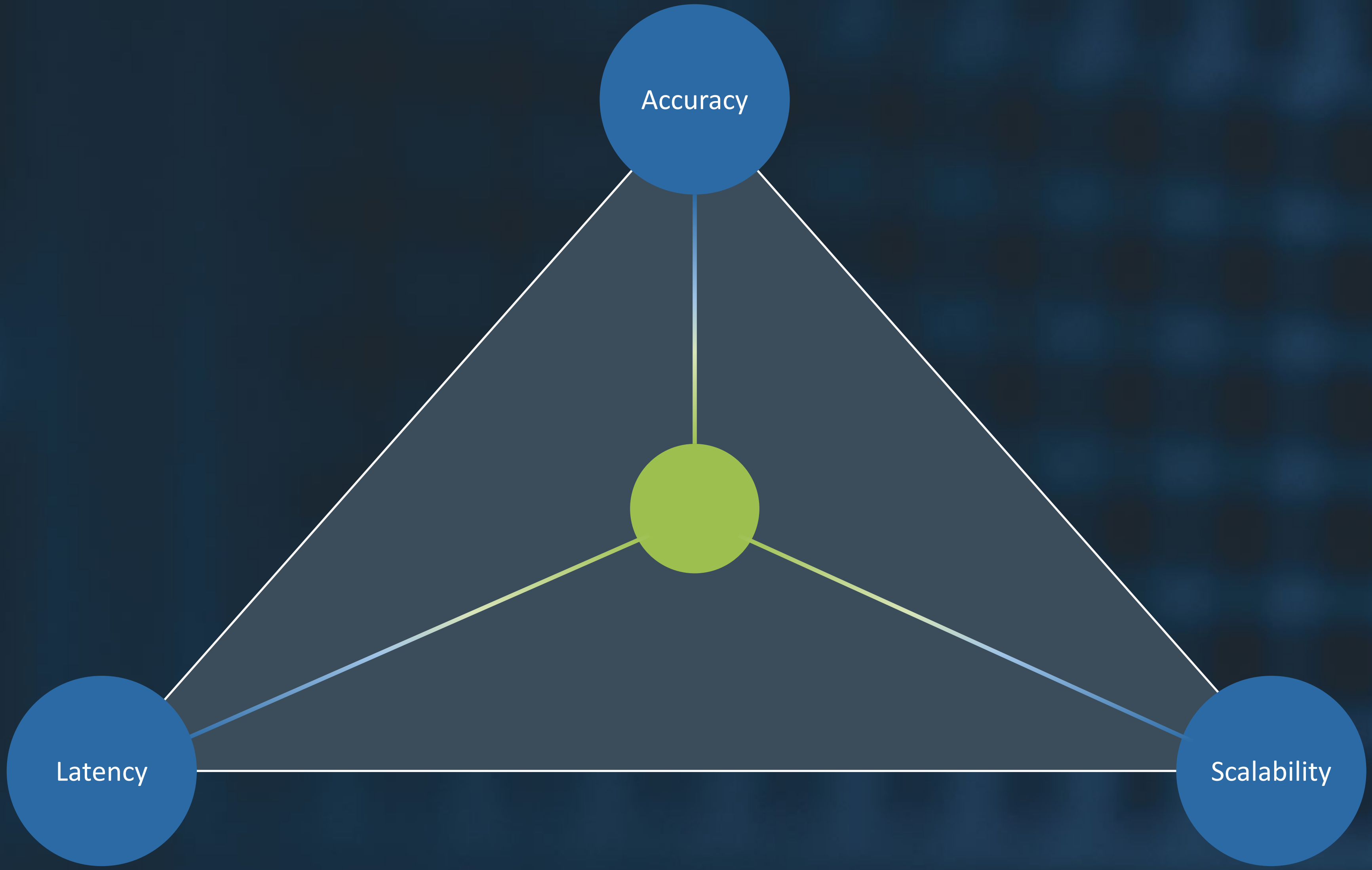
Using temporal segmentation mechanism to make decision

# Did We Do Better?

- Yes!
- Accelerated the performance by 35X
- As a results HW and Investment costs are down by 35X

# Is it Enough?

# Transcription Trade offs



# Personalization
# Speaker Verification

# Speaker Verification

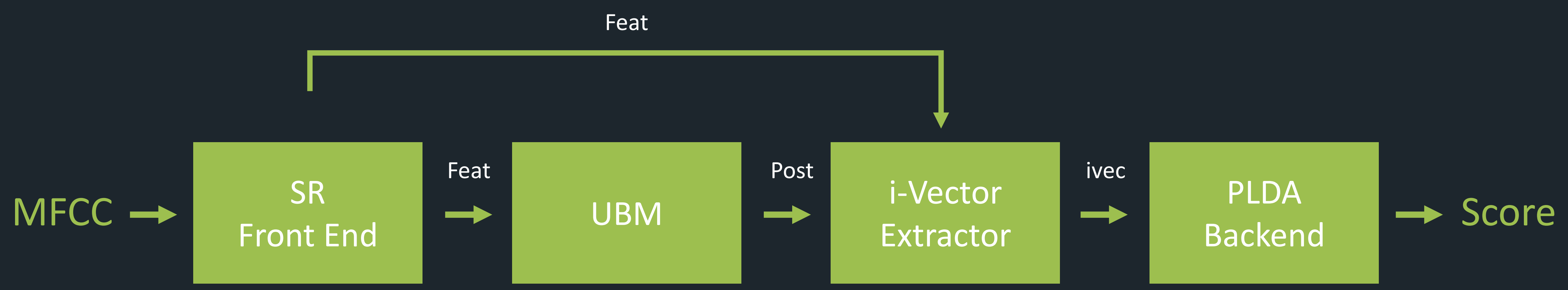
- **Financial institutions lose $10B year due to call fraud**
- Verify if the person who talks is actually the user
- Prevents fraud both for users and agents
- Save a lot of time for the agent and also improves the customer experience.
- Should be **text independent**

# What is the difference between theory and practice

- We need to minimize the time it takes to verify a person
- Anything above 30s is not relevant
- Different noises within the call
- Confidence measures, how sure are we about the hypothesis.

# Proposed Solution i-vector system

- Using an i-vector system
- i-vectors are low dimensional speech representation models
- This is state of the art for most speaker verification methods
- Data was very noisy, so we developed a music and noise detection algorithm (MUSAN)
- Developed an online system



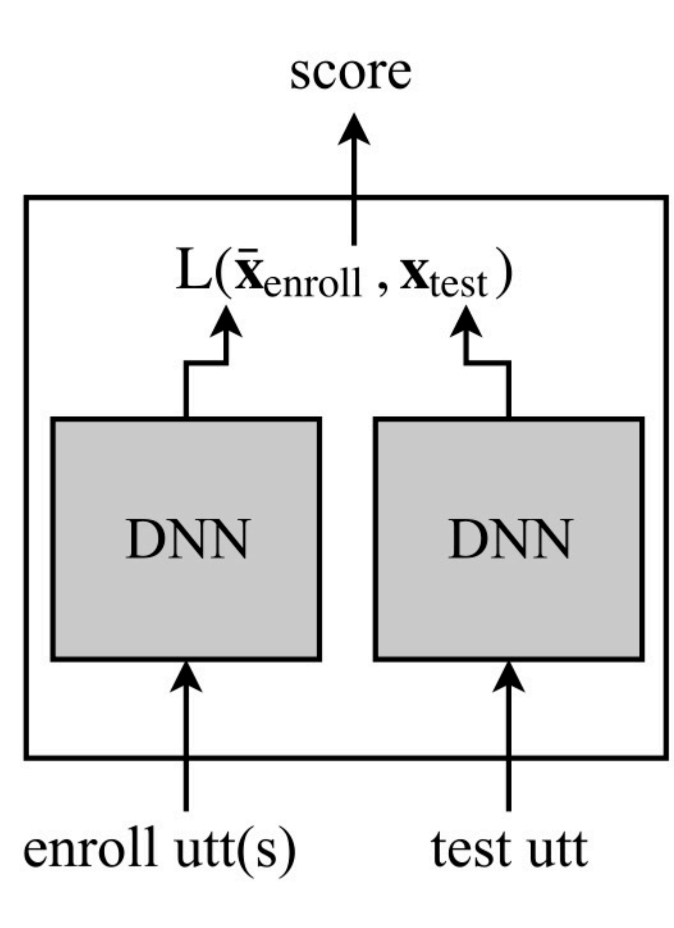
# Reducing the verification time

- For practical applications reducing the verification time is crucial
- An i-vector is extracted at each time step
- Setup a **confidence measure** if to move forward or setup a decision
- **Results:**
  - **2% EER – 98% accuracy**
  - **Average verification time 4.5s**
  - **Median time 2.5s**

# Moving Forward – Speaker Embedding's

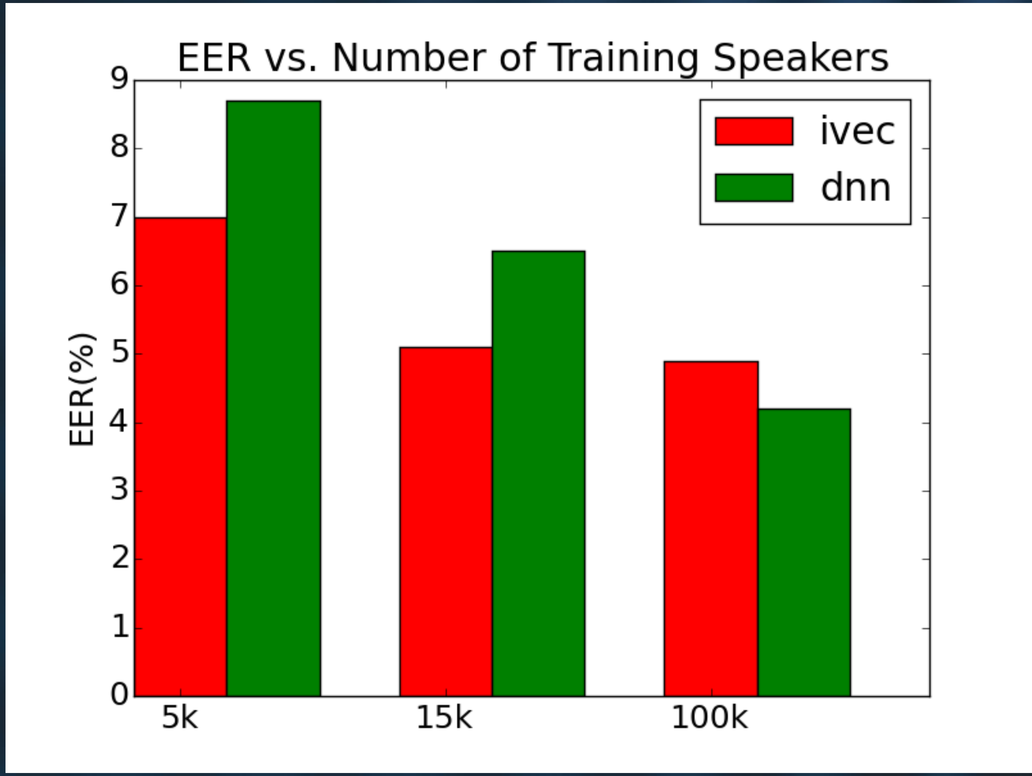
"Deep Neural Network-based Speaker Embedding's for End-to-end Speaker Verification" D. Snyder et al.

- Created an embedded mechanism
- Objective aim – maximize same speaker, minimize different speakers
- Enrollment utterance(s) are mapped to embedding's x_enroll
- Test utterances is mapped to embedding x_test
- Pairs of embedding's are scored using a distance metric L(x, y)



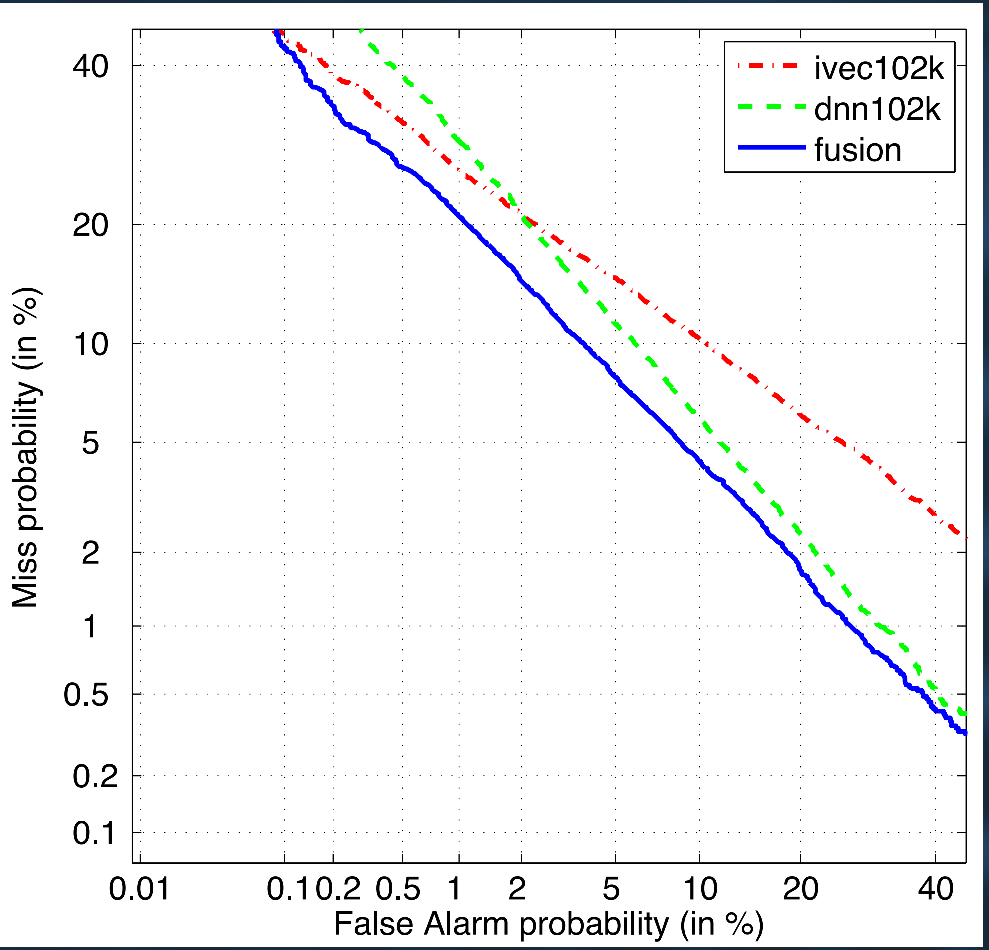
# The importance of large dataset

- NN for speaker embedding's requite lots of data
- We evaluated it on a dataset of 250,000 unique anonymized users
- NN converge and give better results the more data we have.
- On short segments speaker embedding's outperforms i-vector

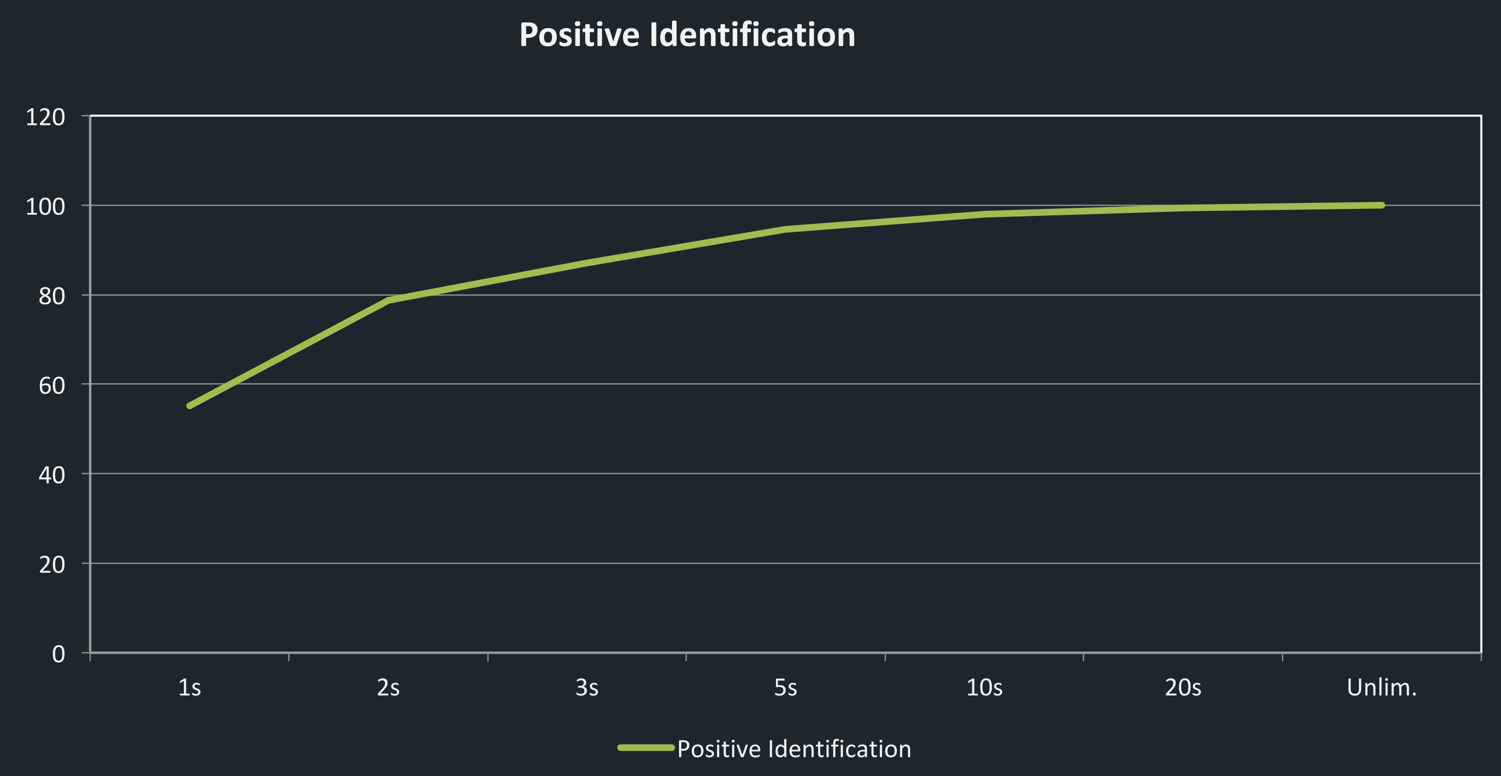


# Fused System and results

- We saw NN system and i-vector system errors are different.
- We created a fused system
- Combining system reduced EER by 30-40%
- **Average time 2.5s Median time 1s**
- **2% EER**



# Operating at Scale




Caller Verification

# The challenges of creating an AI product

# AI Productivity


## Algorithms

Build better algorithms using machine learning and deep learning models

## Data

Use dedicated data to build better models, especially data driven ones (machine learning)

# AI Productivity


## Algorithms
Build better algorithms using machine learning and deep learning models

## Data
Use dedicated data to build better models, especially data driven ones (machine learning)

## Real Time
Optimize algorithms, SW and performance to minimize the latency

# AI Productivity


## Algorithms
Build better algorithms using machine learning and deep learning models

## Data
Use dedicated data to build better models, especially data driven ones (machine learning)

## Real Time
Optimize algorithms, SW and performance to minimize the latency

## Scale
Use clusters, GPU's, parallel algorithms, HPC, micro-services to make sure solution is scalable.

## Product
Wrap everything into a product ready solution, product managing and offering, make sure everything is working from DevOps perspective

# Spōken

www.Spoken.com
yishay.carmiel@spoken.com
@YishayCarmiel