LumenVox®

Deep Neural Networks in Speech Recognition

Why, What & How: End-to-End New DNN ASR Engine

Introduction

This white paper is intended to educate technical audiences on the 'why', 'what', and 'how' of a new end-to-end Deep Neural Networks (DNN) based ASR engine from LumenVox

- Why LumenVox made this investment is key to see the **value of adopting the new engine**
- What we created will expand your thinking as to what **use cases we can now support together**
- How we did it and how you will incorporate it into your business will allow **development and delivery of more value to our shared end-customers**

What you will learn about our new engine's architecture using Deep Neural Networks for ASR transcription:

- A general overview of its benefits and workings
- What has been implemented: details of new capabilities
- System architecture
- Results of performance testing comparing our new engine with:
    - Major Cloud providers' and alternative capabilities
    - Our own legacy system

Benefits of the New Engine & Architecture

- Significant improvement in accuracy
- Faster cycle for supporting new dialects and accents, without needing to add new acoustic models
- Faster cycle for new language support
- Ability to handle lengthy transcription
- Increased transcription speed
- Reduced footprint in storage and memory without degradation in speed and accuracy

- Continued support for customer investment in grammars

In the following sections we review the capabilities that make these benefits possible. For each capability you will learn which of the benefits it provides.

Highlights of New Engine Capabilities

This section highlights the capabilities of the new LumenVox ASR engine. Subsequent sections drill down into the details of each capability:

- **End-to-end acoustical modeling** with Deep Convolutional Neural Networks (Deep CNN). This extends the benefits of neural networks from the sound-recognition stage all the way to the production of text, removing the need for separate acoustic models to accommodate dialects and accents
- **Transfer-learning techniques** from existing models to new models, greatly improving learning efficiency
- Use of quantized **Statistical Language Models (SLM).** Efficiently compressed in storage, quantized SLMs enable higher performance in less memory
- Performance and accuracy aided by traditional **SRGS grammars**
- **A streaming model** that operates online, thereby boosting responsiveness and performance

End-to-End Acoustical Modeling with Convolutional Neural Networks (CNN)

Our Deep Neural Network (DNN) engine uses a Convolutional Neural Networks (CNN) implementation. CNNs are a more advanced algorithm where each node in the network layer learns from select pattern-related nodes in the previous layer, not from the entire previous layer. Thus, a CNN approach develops patterns in successive layers, each building upon the layer before it. These patterns contribute to the learning of the individual words.

In the visual-pattern recognition world, CNN starts by recognizing dots and small areas on the screen. Then it builds on these items to 'see' lines and curves. Finally, it builds on lines and curves to identify shapes.

In the speech recognition world, CNN does something similar: it 'hears' and recognizes small bits of sound; then it builds on sounds to recognize phonemes. Finally, it builds on phonemes to identify words. In contrast, the legacy model does the following:

1. ASR produces a series of phonemes (specific pronunciations).
2. An acoustic model built as a separate step is then used to make sense of these phonemes and build them into words.

The legacy approach may work until the engine encounters a new dialect or a new accent… or sometimes just idiosyncrasies in the way a person talks. At that point, it fails because this new phoneme does not exist in the acoustic model, so we don't know how to map it to a word. Under the old model, every time the dialect or the accent changed, a new acoustic model was required. That can stop customer service in its tracks.

With the new speech engine, the **end-to-end** neural net folds in the problem of dealing with different pronunciations. In other words, we let the most expert system—the neural network itself—handle the problem of variations in speech. With sufficient training data, many different pronunciations can map to the same word automatically. The new architecture allows for incorporating high volume of training into the model. When there are more examples, the recognition is more accurate. Additionally, with the new architecture there is no limit to the number of variations in accent that can be included. This means that the new engine can cover a wide variety of dialects and accents in one extensive model.

Note that governing the words generated, ensuring they are put correctly into phrases and sentences is a separate, subsequent task. This separate task uses a language model (or alternatively a closed-set grammar – a more strict, limited model) to determine if the word identified is indeed the most likely next word. This linguistic task is still performed within our engine, at the end of the acoustic task.

Key takeaway: end-to-end deep CNN acoustic modeling provides for the following benefits:

- Significant improvement in accuracy
- Faster cycle for supporting new dialects and accents, without needing to add new acoustic models
- Ability to handle lengthy transcription

Transfer Learning Techniques

Transfer learning is a machine learning technique in which once a model has been trained for a specific task, it is then reused to initialize a model for a different but related task. This is a common approach for deep learning where pre-trained models are used as a starting point.

In our case this means, for example, that we can reuse the acoustic model trained on English—for which there are thousands of hours of training available—for Italian, for which there is a reduced amount of training. There is enough similarity between the two languages for training to overlap.

This results in better accuracy and performance compared to training the Italian acoustic model from scratch. Furthermore, transfer learning speeds up the training process considerably (days versus months). This lowers the cost of training models for new languages.

Key takeaway: transfer learning techniques provide for the following benefits:

- Significant improvement in accuracy
- Faster cycle for new language support

Quantization of Sorted-Tree Statistical Language Models (SLM)

We previously discussed the linguistic task of verifying how words are put together, that follows the acoustic task of recognizing the words. The linguistic task is governed by grammatical rules. These grammatical rules are specified in the language model (or in a custom grammar if used). The more complex the language, the more rules are contained in the language model. Storing the more complex language models, if inefficient, affects performance and speed of service when it takes up a great deal of memory.

With the technology used by the new engine, language models are stored using binary trees—data structures that are searched with O(log n) complexity, which is particularly good for large numbers. The new engine also uses a compression scheme that allows larger and deeper language models to be stored in a smaller space. More detailed language models are therefore represented in less space, which means they are faster and more efficient, doing more with less memory.

Key takeaway: quantization (sorted storage and compression) of SLMs provides for the following benefits:

- Increased transcription speed
- Reduced footprint in storage and memory without degradation in speed and accuracy

Supporting Use of Traditional SRGS Grammars

SRGS (Speech Recognition Grammar Specification, a W3C standard) grammars, written in either BNF or XML style, specify patterns of words that are expected to appear together and the order in which they can appear. Grammars may be used as part of the linguistic task, as an alternative to a complete language model. They help predict what combination of words are likely, in the case of ambiguous or unclear pronunciations, so they are used as part of the speech recognition process.

Such grammars are typically used by IVRs (Interactive Voice Response) to also interpret combinations of words to recognize a limited set of possible meanings. For example, “yes,”

"yes, please," and "okay" are all interpreted as "yes." This means that they are used not only for speech recognition, but also, in a sense, for semantic interpretation.

The LumenVox legacy system had the ability to use such grammars in lieu of using a complete language model, for customers and applications that only expect such limited words and phrases.

Speech recognition of a limited set of potential words is always more accurate than having the entire language in the pool of candidate words. This works very well in applications where users don't expect the freedom to say whatever they want, but rather they know they have to stick to limited options. The result is that when using grammars, the speech engine makes fewer errors.

The new engine not only retains this capability, and can use these grammars in the same way, but also greatly improves it. Using the new learning capability enables significantly more accurate processing of sounds into text, when using grammars. It can bring down the WER (Word Error Rate) in some cases to below 1%, as you will see in the performance testing section.

Key takeaway: supporting the use of traditional SRGS grammars provides for the following benefits:

- Significant improvement in accuracy
- Increased transcription speed
- Continued support for customer investment in grammars

A Streaming Model

The new engine makes direct use of speech input in real-time. Incoming sounds are processed as received, and knowledge picked up during one pass is utilized right away. As a result, there are no multiple passes or extra-cycles of processing, and there is no latency or perceptible delay between the caller speaking and the system responding. The result is a system that is faster and more responsive.

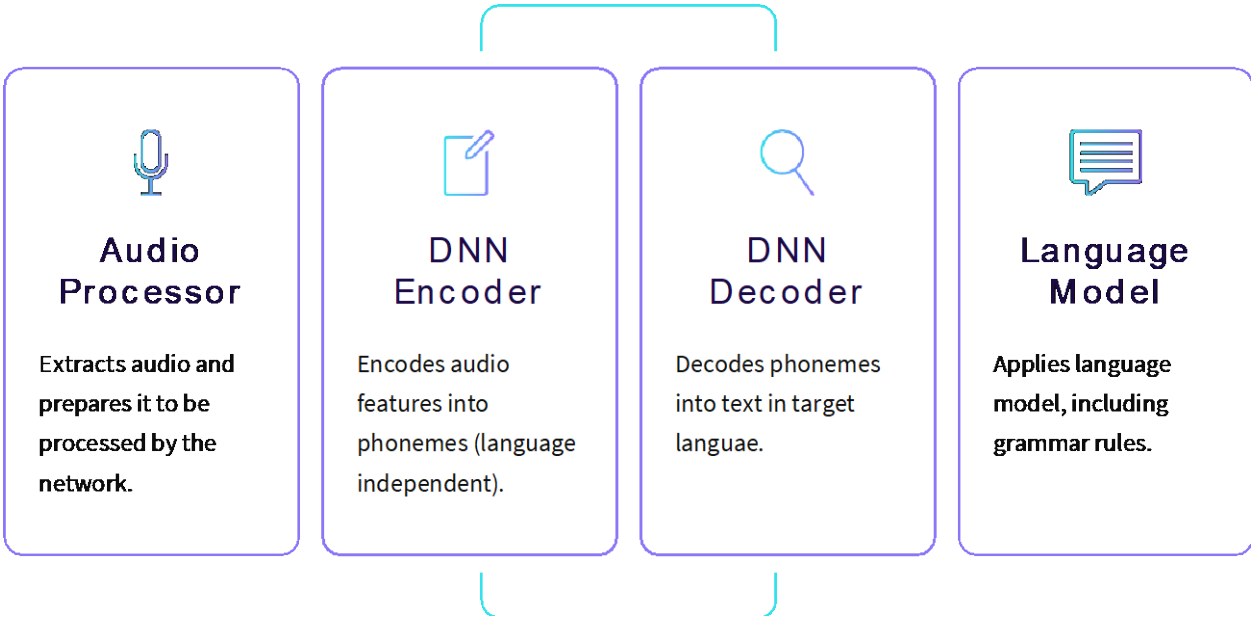
Key takeaway: a streaming model provides for the following benefits:

- Ability to handle lengthy transcription
- Increased transcription speed

Engine Architecture

The block diagram on the next page shows the architecture of our engine. It consists of three main blocks, where blocks 1 and 2 perform the acoustic task, and block 3 performs the linguistic task

1) **Audio preprocessor.** It extracts features from the audio, such as Mel-Frequency Cepstral Coefficients (MFCC), and prepares them in chunks, to be processed by the network.
2) **Deep Neural Net (DNN).** This block processes the input audio features and generates sequences of characters (words) in the target language. Internally, it consists of two parts:
    a) An **encoder** that encodes the audio features into an internal phonetic representation. This is largely language independent.
    b) A **decoder** that transforms this internal phonetic representation into target language characters. This is language dependent.
3) **Language model.** Applies the restrictions enforced by the desired language model (or traditional closed-set grammar) to the network output.



Workings of Block 2: End-to-End DNN

Legacy HMM Falling Short

Traditionally, speech recognition engines rely on Hidden Markov Models (HMM). HMM algorithms implement statistical analysis using the behavior of an **external** factor to predict the behavior of an **internal,** hidden factor. Speech recognition engines look at the external data—the sounds produced—to predict the words intended by the speaker.

HMM is an older algorithm that was found useful in many fields. However, legacy HMM based ASR ended up being too complex and inefficient. It is definitely machine learning, but it isn't end-to-end.

In the legacy architecture there is a disconnect between the process of creating phonemes from the audio, and the process of mapping the phonemes into words. The mapping is done using separate, manually-maintained lexicons, and linguistic resources such as a pronunciation dictionary, tokenization, and phonetic context-dependency tree. These lexicons are specific to dialects and accents. It is not possible to train the lexicon for more than one dialect at a time. This introduces limits to lexicon size, and also to the size of training data that was used to create them. As a result, they fall short in modern times in performance, both in speed and accuracy.

Modern DNN: Training and Benefits of an End-to-End Approach

The next generation of technology is "Deep Learning," implemented through Deep Neural Networks (DNN). In recent years major technology advancements have been made by Computer Science researchers in both Speech Recognition, as well as Computer Vision fields. At LumenVox our science researchers have created a modern machine learning system that incorporates all of the required activity that builds an acoustic model into a single, self-improving, deep layered network architecture. It is a single entity, despite the fact that for illustration purposes the diagram above shows it as 2 parts within Block2. Our architecture creates an extensible acoustic model that doesn't impose limits on the number of dialects and accents that can exist for each word and does not require separate lexicons and other resources to produce an accurate output.

An end-to-end DNN system involves more training than HMM systems, and that training is put to good use. Training is the process of machine learning: understanding probabilities in training data, to be able to infer from them conclusions about new data. It's all about math, and DNNs use a lot more data and do a lot more math as part of the training process.

The neural net is initialized when it's fed a large sample of raw data and corresponding answers. Based on probabilities present in the data every node in the network gets a value, and a computation is made to get an output and compare it to the answer. Initially the output might be far from the right answer. The network uses linear algebra to implement back

propagation to move toward more correct pattern-recognition behavior after each trial. Node values are revised in each trial. This process may be repeated thousands of times. In the end a model is created that provides accurate results, where the output is consistently close to the correct answer. The model is then evaluated on test data, which was never used in the training process, again comparing the output with correct answers. If it passes successfully, the model is now ready to be fed customer data and output the expected transcription.

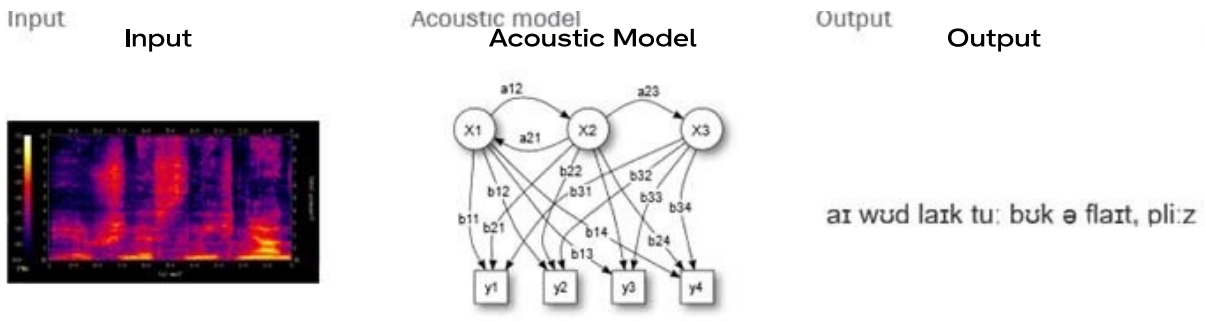
This architecture is able to avoid the use of a separate acoustic lexicon because there is no need for a 'source of truth'. The truth is learned from the training data, and many truths are allowed to co-exist by allowing many inputs (audio bits) to map to an output (text) without intervening or limiting the mapping to and from internal phonemes, if the data says they should map. In essence we are replacing the use of linguistic resources with a data-driven learning method. Our state-of-the-art deep-learning system achieves with this end-to-end design significantly better results, that get even better with additional training, both in speed and accuracy.

CNN: Adding a Twist to the Secret Sauce

As mentioned earlier, our DNN engine has a convolution twist. Convolutional Neural Networks (CNN) provide a still deeper level of analysis, building up levels of comprehension and recognition. More specifically, the CNN convolutional process recognizes smaller patterns, then recognizes phonemes, and finally builds upon those smaller objects to produce words. CNNs have been used in the past decade for computer vision, and recently have also been introduced to audio processing, but using 1-D convolutions (in time domain) vs. the 2-D convolutions commonly used in image processing. More reading is recommended outside this white paper, for those interested in understanding CNNs in depth.

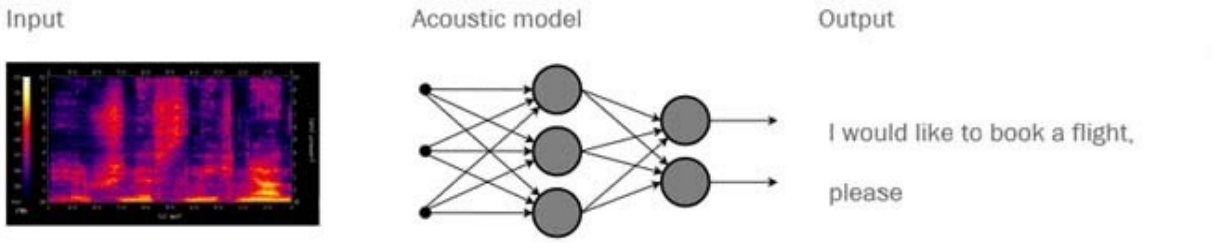
Acoustic Modeling: Old vs. New End-to-End System Illustration

The following graphic shows how the **legacy HMM Kaldi-based engine** analyzes sounds. These need to be analyzed by a different acoustic model for each dialect.



In contrast, an end-to-end speech recognizer links audio utterances and resulting text directly. With our new ASR engine, we do not need to use a phonetic dictionary (or lexicon) to transcribe the training text material. The engine itself handles interpretations of phonemes internally.

The following graphic shows how much more efficient and accurate this process is, as it uses our **new end-to-end neural networks acoustic modeling** to take sound and produce words of text as the output in one step.



For each language for which we train the network, we leverage hundreds or thousands of hours of audio, covering dialectal variations and environmental conditions. This allows us to train a single model for one language instead of having to train individual ones (e.g., one English model would cover US, UK, AU, etc.). Given enough examples, the system will learn to recognize the word **"tomato"** for both US pronunciations **("təˈmeɪˌtoʊ")** and UK pronunciations.

End-to-end systems can also be fine-tuned for specific conditions, such as telephone channel or noise. This can be done by obtaining such noisy audio data or creating audio noise in an artificial manner for training purposes. It should be noted that given the very good generalization performance of the acoustic model, it is usually enough to just fine-tune the model with more general training.

Language modeling: End-to-End DNN Applied to Block 3

We discussed how we build our acoustic models, but what about language models? In contrast to acoustic models, building a language model has nothing to do with audio. It is all about understanding a language from its building blocks, and that can be done completely by using just the text.

The good news: our engine provides out-of-the-box statistical language models trained using millions of text lines from many sources, accounting for several use cases: generic transcription, spellings, digit string recognition, etc.

What we found is that deep learning systems benefit from more programmed expertise up-front. As it turns out, this has been a worthwhile investment. We reap major benefits by using our neural net technology to handle more of the overall process including the language modeling. It is thus truly "end to end."

Customizing the Language Model

We offer several options to customize the language modeling part of the recognition:

- Adding lists of words or phrases, which can be done on a per-request basis or caching such requests; this feature boosts the recognition of the specified words or phrases. A typical use case would be adding a list of employees that need to be recognized or specific product names (e.g., pharmaceutical drugs or specific restaurant dishes).
- Adapting the standard language model with a small amount of domain-specific text to boost recognition of in-domain audio. This is for those situations where there is not enough text data to fully train a new language model, but enough so that recognition of domain-specific text is enhanced. Performance Testing Results

In the following section we review results of performance testing conducted to compare our new engine to the competition.

1. The **transcription use case** of our engine is compared to:
    a. Major Cloud providers' capabilities
    b. Alternative Kaldi-based engine capabilities
2. The **closed-set grammar use case** of our engine is compared to
    a. Our own legacy system grammar capabilities

Transcription Performance Testing

Comparison with Cloud Providers

English: Accuracy

Engine/ Dataset	Amazon Transcribe	Amazon Lex	Google Speech-to-API	Microsoft Cognitive Services	LumenVox ASR Transcript	LumenVox ASR Grammar
CSLU Digits	94.3%	91.5%	89.3%	98.4%	98.5%	98.7%
CSLU Words	37.1%	31.1%	41.5%	62.1%	58.8%	98.7%
CSLU Phrases	71%	66.5%	73.9%	86.3%	84.6%	98.6%
Macrophone	Not possible to normalize results	93.4%	Not possible to normalize resutls	Not possible to normalize results	95.4%	Not applicable to grammars

Note: This table is showing word accuracy. Higher values indicate better performance.

The above table describes tests conducted using 4 different datasets. As you can see, the performance advantage for the new LumenVox ASR engine is substantial. The transcription engine performed on par or better than the rest of the transcription services evaluated. When a grammar was used instead of free-form transcription, for data sets where a closed grammar can be reasonably defined for the vocabulary, then the accuracy was phenomenal.

Comparison with Kaldi-Based Engine + Models Created on Kaldi Platform

English: Word Error Rate

Engine/Data-set	LumenVox ASR Transcritp	Kaldi: UK Eng Model	Kaldi: US Eng Model	Kaldi: US Eng + Robust Model
CSLU Multilangual	17.4%	32.2%	17.6%	21.2%

Note: This table is showing word error rate (WER). Lower values indicate better performance.

This table describes performance tests comparing the LumenVox engine and LumenVox English acoustic model against multiple acoustic models that were created using the Kaldi platform, and then run with the Kaldi-based engine.

- Column 3 from the left shows results for a British English Kaldi model (en_GB-8kHz)

- Column 4 from the left shows results for a US English Kaldi model (en_US-8kHz)
- Column 5 from the left shows results for a robust US English Kaldi model (en_US_robust-8kHz)

The tests were conducted using the CSLU Multilingual (English set): landline speech, part scripted, part free, short utterances. As we can see, our new engine performs better than the Kaldi based engine and models.

Spanish: Word Error Rate

Engine/Dataset	LumenVox ASR Transcritp + SLM	Kaldi Engine + Model
Mexican Sala II	16.2%	24.8%
Voice Across Hispanic America (VAHA)	9.8%	19.3%
Multilangual Librispeech Spanish	15.7%	32.2%
Multilangual TEDX Spanish	21.3%	32.8%
Mozilla Common Voice Spanish	17%	35.7%
Voxforge (random sub-set of whole corpus, may overlap with training data, thus the unrealistically good results here)	4.6%	0.9%

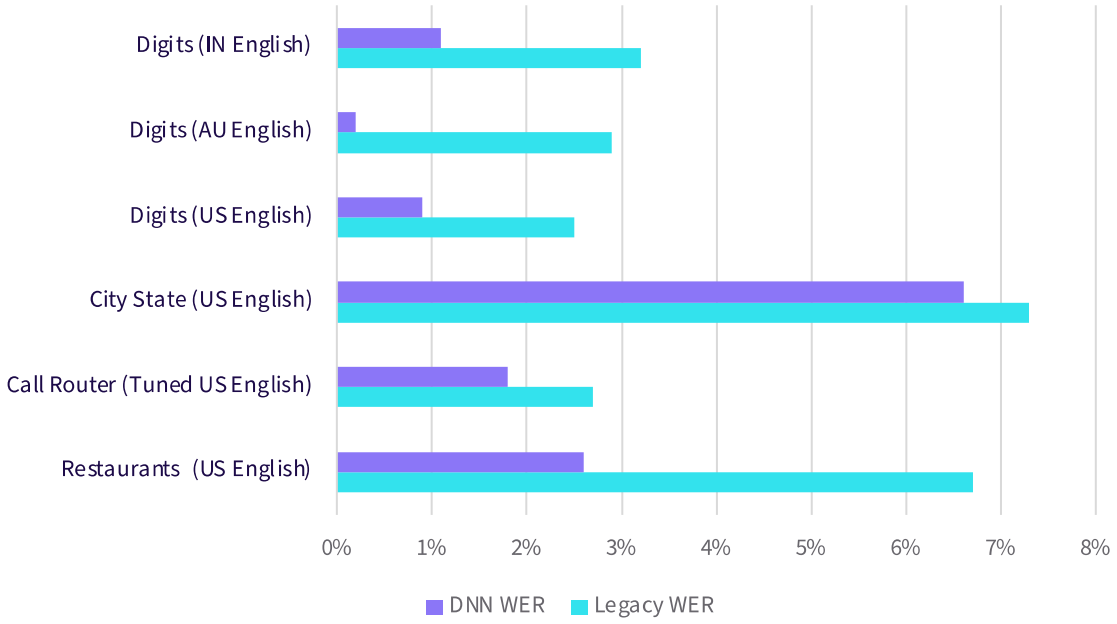
Note: This table is showing word error rate (WER). Lower values indicate better performance.

For Spanish we have evaluated both our new engine with our SLM, and the Kaldi-based one with a Kaldi-generated model, on several datasets. Two of them commercial telephony datasets (SALA II and VAHA), and four of them open-source, non-telephony datasets.

Our new engine performs well on all datasets, whereas the Kaldi-based performance drops especially for the open-source datasets. According to the internal README provided with Kaldi models, their engine is trained mostly in telephony speech and may be limited in terms of generalization to different channels and environments.

Grammar-Based Performance Testing

Comparison with our Legacy Engine



Note: This table is showing word error rate (WER). Lower values indicate better performance.

Performance tests were conducted using tests from the legacy ASR suite to handle grammar parsing and custom pronunciations. In these tests the new engine using SRGS grammars, was compared with the legacy engine.

The result: the new engine outperformed the legacy ASR on all the tests, achieving very low error rates. Furthermore, for the digits use case we show how good our model generalizes to the different dialectal variations (US, AU, and IN English dialects).

Conclusions

- Technologically, there is no turning back. The new ASR delivers a level of accuracy in word recognition that the legacy technology cannot match.
- Legacy systems may perform well enough in the short run, but deep learning with CNN has the advantage of greater knowledge placed into the network up front to the benefits of customers and users. The result is a more expert, and more accurate, speech recognition engine.
- The performance of legacy HMM models tend to be capped at a certain level; beyond that additional training produces diminishing returns. Our new Deep CNN engine can take in orders of magnitude more training and only get better.
- Our state-of-the-art ASR engine also benefits from other technological improvements, such as compact storage of search trees and one-pass technology. Such improvements create a better performing, more responsive system without noticeable latency

End-to-end deep learning eliminates reliance on post-processing of phonemes to do the final translation into text. Therefore, a single DNN based ASR can handle a large domain of input. There is no need for a separate model for dialects and accents. This is a significant benefit for partners, customers, and end-users alike.

LumenVox

About LumenVox

LumenVox transforms customer communication. Our flexible and cost-effective technology enables you to create effortless, secure self-service and customer-agent interactions. We provide a complete suite of speech and authentication technology to make customer relations faster, stronger and safer than ever before. Our expertise is extensive— we support a multitude of applications for voice biometrics, inclusive of passive and active authentication for fraud detection. And we do it all by putting you and your customers first.

Interested in finding out more about this product?

Learn More →

Contact

LVsales@lumenvox.com +1 (858) 707-7700

www.LumenVox.com