THE MARCHEX 2018 ENGLISH CONVERSATIONAL TELEPHONE SPEECH RECOGNITION SYSTEM

Seongjun Hahm, Iroro Orife, Shane Walker and Jason Flaks

Marchex Inc., 520 Pike, Seattle, WA 98101, USA
{shahm, iorife, spwalker, jflaks}@marchex.com

ABSTRACT

In this paper, we describe recent improvements to the production Marchex speech recognition system for our spontaneous customer-to-business telephone conversations. We outline our semi-supervised lattice-free maximum mutual information (LF-MMI) training process which can supervise over full lattices from unlabeled audio. We also elaborate on production-scale text selection techniques for constructing very large conversational language models (LMs). On Marchex English (ME), a modern evaluation set of conversational North American English, for acoustic modeling we report a 3.3% \(\{\text{agent, caller}\}:\{3.2\%, 3.6\%\}\) reduction in absolute word error rate (WER). For language modeling, we observe a separate \{1.3\%, 1.2\%\} point reduction on {agent, caller} utterances respectively over the performance of the 2017 production system.

Index Terms— conversational speech recognition, acoustic modeling, language modeling, semi-supervised training, data selection

1. INTRODUCTION

Marchex’s call and speech analytics business handles over one million calls per business day, analyzing decades of audio per week. These spontaneous conversational, consumer-to-business phone calls occur on modern mixture of mobile phones and landlines, capturing everyday North American dialog in every possible accent variant, speech rate, English language fluency, speaker demographic, under broad environmental conditions, with a comprehensive, colloquial vocabulary. Although there have been incredible performance improvements on large vocabulary continuous speech recognition (LVCSR) task \[1\] \[2\] \[3\], these variabilities, which vary over time, often obscure the true performance of a production automatic speech recognition (ASR) system due to inconsistencies between the training and test set.

To more fully harness the scale of Marchex call traffic, this paper elaborates on the semi-supervised approach that we introduced in \[4\] to substantially increase the quantity and quality of telephone speech transcripts available to train a modern, production-ready conversational LVCSR system.

For acoustic modeling, it is valuable to have such a large and varied dataset which captures diverse language contexts, noise conditions as well as changes in acoustic channel features based on shifts in device and codec technology. To obtain better quality transcripts, we re-decoded the original unsupervised dataset of 30,000 hours of audio, using the production online-nnet2 model. Post-processing and re-selecting the most suitable utterances yielded a new 5,500 hour dataset that was used to train a series of time-delay neural network (TDNN) acoustic models with the LF-MMI sequence objective function \[5\] \[1\] \[6\].

For language modeling, we build on the existing text transforms created to correct systemic decoder errors \[4\]. We dramatically scale up the candidate text we consider for language modeling by aggregating all production utterances. By correcting systemic errors, then filtering by perplexity, utterances with a minimum length, from over a billion utterances, we were able to capture some 9.4M unique high quality utterances with long temporal contexts. These LM text improvements independently lowered WER by an additional point on both caller and agent channel.

Finally, our objectives in this work is to obtain the lowest possible WER production system on the ME evaluation dataset. Unlike other efforts \[6\] \[7\] \[1\] \[2\] \[3\] that explicitly eschew practical considerations such as speed (decoding real-time factor) and simplicity, we have concentrated on making the best use of our data scale to efficiently improve our production ASR accuracy. The importance of ASR accuracy with respect to downstream natural language processing (NLP) model performance can be understood in two ways. 1) Some types of mis-transcriptions characterized by admissible substitutions can be learned by downstream classifiers. To the extent that the sequence is not distinct from other conversational utterances, precision will suffer. 2) Mis-transcriptions characterized by deletions, especially where monosyllabic words are key features, lead to downstream models that learn spurious relationships \[8\].

The paper is organized as follows, Section 2 will provide an overview of the speech recognition system, detailing the updates to the acoustic model training regime as well as data selection approaches for very large language models. Section 3 will summarize the experiments and results while Section 4 provides perspective on lessons learned and directions for future work.

2. SYSTEM IMPROVEMENTS

Marchex call processing servers receive two channels of 8kHz \(\mu\)-law encoded audio for the caller and agent call-legs. Both channels are individually segmented by an online Voice Activity Detector (VAD), creating single utterances that are stored in a large Apache Kafka \textit{audio topic}, each transcribed in turn by a fleet of Amazon Web Services (AWS)-based production Kaldi \[9\] hosts. Each Kaldi hosts runs a decoder using a deep neural network acoustic model and utilizes a decoding graph with an enhanced lexicon and a very large, conversational language model (LM).

In preparation to improve the existing \textit{online-nnet2} training regime, the lexicon was enhanced with pronunciation probabilities for words with multiple phonetic pronunciations and silence probabilities between phonemes within a word were also added \[10\]. Next, acoustic model training recipes were updated to use Kaldi \textit{nnet3} TDNN architectures \[5\] \[1\] \[12\].

The tools and training recipe for our experiments are based on
the semi-supervised LF-MMI work by Manohar et al. [11] who use the Fisher English [13] dataset with a seed model trained on purely supervised (only hand-labeled) transcripts. By contrast, our TDNN seed model was trained on semi-supervised data (5,500 hours). Because our seed model starting from an already semi-supervised dataset, we expect to reduce WER considerably.

2.1. Semi-supervised acoustic modeling

A TDNN is a kind of feed-forward neural network architecture shown to be effective in handling long range temporal dependencies. Each layer operates at a different temporal resolution with initial layers processing smaller contexts, while deeper layers attend to wider temporal contexts. TDNNs improve on limitations of traditional recurrent neural networks (RNNs) i.e. high computational complexity that is non-parallelizable, with a sub-sampling technique where each layer’s input is selected from specific time steps in the previous layers. By carefully selecting splicing indices, per-layer computation is reduced while assuring an adequate amount of temporal context is seen by each layer of the network [10, 5].

In moving to new nnet3 TDNN recipes, a number of updates were made along the way. These included overhauling the non-linearity function from p-norm to ReLU, applying dropout and tuning the size of the left/right context and the dimension of the output layer. The final recipes also utilized a factorized TDNN or TDNN-F from [12], which improves on the state of the art TDNN+LSTM (long short term memory), by constraining one of each weight matrix’s two factors to be semi-orthogonal. TDNN-F also employs a ResNet-style skip connection to concatenate the output of previous non-adjacent layers.

2.1.1. LF-MMI Training

We will give a brief overview of LF-MMI training with semi-supervised lattice supervision. For a full description, the reader is assigned an exhaustive study of [9, 10, 5]. For acoustic modeling, maximum mutual information (MMI) is an objective function used in sequence discriminative training of neural networks. Word lattices are generated from the training data using a frame-level, cross-entropy pre-trained Gaussian mixture model (GMM) model and a “weak” language model. There are two sets of lattices, a numerator and denominator lattice. The numerator represents the “alignment” of the correct sentence, while the denominator lattice embodies all possible word sequences from the recognizer and is generated using a very big beam. The denominator lattice in the training objective is the same for all utterances. Optimization involves gathering statistics and re-estimating model parameters using the forward-backward algorithm over these word-level lattices. As elsewhere, the training objective is to maximize the conditional log-likelihood of the correct transcript.

In LF-MMI, sequence discriminative training is done from “scratch”, at the finer phone resolution, without using word-lattices. A GMM or deep neural network-hidden Markov model (DNN-HMM) system may be used generate lattices from which phone graphs are derived. For the MMI objective, there are now numerator and denominator graphs which are stored as Finite State Acceptors (FSA). Accordingly, there still needs to be a summation over all possible label sequences, so a 4-gram phone-level LM, in lieu of a word-level LM, is used to create the denominator graph, which is post-processed to be as small as possible for practical on-GPU training.

2.2. Data selection for very large language models

For language modeling, we have traditionally relied on some 14,000 hand-labeled utterances, customer text from onboarding, perplexity-filtered, semi-supervised text labels also used for acoustic modeling (Figure 1) as well as 6,300 hand-curated, high-value, high-
prevailing usages are more common during customer-to-business telephone calls. Because we are able to capture live production traffic, taken directly from our production Apache Kafka text topic, affectionately dubbed the “firehose” (FH), we were able to build a completely new, raw text corpus. Decoding on average 500M utterances per month, this FH text corpus ballooned to over a billion spontaneous conversational utterances at the time of our experiments, outlined in Table 1. Aside from the text selection and filtering algorithms previously described in [4], the additional selection heuristics used in this work are as follows:

- **min-utt-len** enforced a minimum admissible utterance length. We experimented with values on the range (3, 20) and only report results for 12 and 6. The latter gave us the best tradeoff of quality vs. sufficient number of utterances to make an impact on WER.
- **sort uniq** picked out only unique utterances. We discovered that at this scale, there was more value in variety than in repeated utterances.
- **ppi** filtered utterances with maximum perplexity of 30, a figure reached via a short grid search. Perplexity was computed with a 5-gram LM trained from hand-labeled utterances (20K) and the highest prevalence, manually corrected utterances (6.3K).

### 3. EXPERIMENTS

To verify effectiveness of LF-MMI lattice-supervision, we performed semi-supervised LF-MMI experiments using 5,500 hours training data. All experiments were performed using Kaldi toolkit [2].

#### 3.1. Experiments on a 550 hour subset

To validate suitability of LF-MMI lattice-supervision, we performed preliminary experiments. A baseline seed model was built on a subset of ME. Table 2 shows a summary of training condition. Utterances were chosen based on the criteria that there be sufficient leading and trailing time, i.e. are correctly segmented. This seed model was trained with 14 supervised hours and 534 unsupervised hours. iVector extraction was also trained on the combined datasets. The lattices generated by decoder were then used to train a new model using lattice-supervision. We trained 4 different kinds of models, TDNN (tdnn.70), TDNN-LSTM (tdnn_lstm.60), bi-directional gated recurrent unit (bGRU; bgru.6), and factorized TDNN-F (tdnn.70).

| LM       | Modeling Text | # Words | WER Δ |
|----------|---------------|---------|-------|
| 3-gram   | Production text | 1.67M   | -/-   |
|          | Firehose (FH)  | +32M    | -1.2/-1.0 |
|          | FH + len12 + sort.uniq | +2.3M | -0.9/-0.9 |
|          | FH + len6 + sort.uniq  | +9.4M  | -1.3/-1.2 |
| 5-gram   | Production text | 1.67M   | -/-   |
|          | Firehose (FH)  | +32M    | -1.8/-1.6 |
|          | FH + len12 + sort.uniq | +2.3M | -1.5/-1.2 |
|          | FH + len6 + sort.uniq  | +9.4M  | -1.9/-1.5 |

Table 3 shows the results. We considered WER, the number of model parameters, the size of acoustic model (AM), decoding speed (real-time factor; RTF) to select the seed model for semi-supervised experiment. Although bGRU has the best WER among four models (first 4 rows in Table 3), we selected TDNN-F (tdnn.70) based on WER (second best) and RTF (best). Then using TDNN (tdnn.70) as a seed model, we performed semi-supervised experiment. The results showed a WER reduction in the new model of {1.9%, 2.2%} on the {agent, caller} utterances respectively.

We repeated these experiments, rebuilding the seed model from 5,500 hours of semi-supervised labels described in the Introduction. Our production model is the output of 2 iterations with the factorized TDNN-F model (tdnn.70). In Section 3.3, we report the results of all experiments.

#### 3.2. Training condition

In Figure 1, the sequence of training operations is outlined, including how the seed for the current LF-MMI training (in green) is built on the already semi-supervised transcripts from [4]. We note that the lexicon includes a standard tokens such as [noise], [laughter] and [unk], as well as two additions: [tone], [spanish], the former to annotate a ring-tone and the latter a first attempt to distinguish common non-English utterances and its dialects. We also note that for the total number of speakers we assume that each channel has only a single speaker, though in reality we see the same agents multiply and customers do call back. Detailed training condition was shown in Table 2.

#### 3.3. Experimental results

Table 4 shows the results of semi-supervised acoustic modeling described in Section 2.1. Our results are based on ME, a modern-day North American English conversational task comprised of 7,000 utterances or 4.5 hours of no-filler, manually transcribed conversational audio, sourced from more than 3,000 calls. We report WER figures, per call-channel as well results from rescoring with a 4-gram LM and a Tensorflow LSTM LM (TF-LSTM).

We followed standard scoring method similar to other research [1, 2, 14]. There were a scoring changes from our previous results reported in [4]. Those are partial words {gues- vs. guess}, multiple words {firestone vs. fire stone}, colloquial forms {going to vs. gonna}. Hesitations and filled pauses {uh, um, ah, er} were also removed. In Table 4, you can see the baseline performance

---

1 We did make small change (putting right side GRU-layer instead of TDNN-layer) based on tdnn_opbgru.1b.

2 The current performance for [tone] and [spanish] is poor due to insufficient number of labels.
### Table 3. Preliminary experimental results using 550 hour subset (WFST: Weighted Finite State Transducer)

| Test set  | AM               | WER % (3-gram LM) | # Model Params | AM Size | WFST Size | Decoding RTF |
|-----------|------------------|-------------------|----------------|---------|-----------|--------------|
| Agent     | TDNN             | 10.4              | 11.4 M         | 44 MB   | 370 MB    | 0.29         |
| Caller    | tdnn_7l          | 14.3              |                |         |           |              |
| Combined  |                  |                   |                |         |           |              |
| Agent     | TDNN+LSTM        | 10.7              | 18.6 M         | 72 MB   | 370 MB    | 0.57         |
| Caller    | tdnn_lstm_lb     | 14.0              |                |         |           |              |
| Combined  |                  |                   |                |         |           |              |
| Agent     | bGRU             | 9.3               | 47.5 M         | 183 MB  | 370 MB    | 2.39         |
| Caller    | bgru_1a          | 12.6              |                |         |           |              |
| Combined  |                  |                   |                |         |           |              |
| Agent     | TDNN             | 9.9               | 22.2 M         | 86 MB   | 370 MB    | 0.24         |
| Caller    | tdnn_7o (seed)   | 13.0              |                |         |           |              |
| Combined  |                  |                   |                |         |           |              |
| Agent     | TDNN             | 8.0               | 22.2 M         | 86 MB   | 370 MB    | 0.26         |
| Caller    | tdnn_7o (semi-sup) | 10.8           |                |         |           |              |
| Combined  |                  |                   |                |         |           |              |

### Table 4. WER performance and model attributes

| Test set  | AM               | WER % (3-gram) | # Model Params | AM Size | WFST Size | Decoding RTF |
|-----------|------------------|----------------|----------------|---------|-----------|--------------|
| Agent     | DNN              | 10.3           | 13.7 M         | 54 MB   | 1.1 GB    | 0.99         |
| Caller    | nnet.2           | 9.8            |                |         |           |              |
| Combined  |                  | 9.8            |                |         |           |              |
| Agent     | TDNN             | 8.3            | 23.4 M         | 91 MB   | 370 MB    | 0.34         |
| Caller    | tdnn_7o (seed)   | 10.9           |                |         |           |              |
| Combined  |                  | 10.2           |                |         |           |              |
| Agent     | TDNN             | 7.1            | 23.0 M         | 89 MB   | 370 MB    | 0.30         |
| Caller    | tdnn_7o (semi-sup) | 9.6           |                |         |           |              |
| Combined  |                  | 9.2            |                |         |           |              |

is 10.3% and 13.2%, agent and caller respectively, based on new scoring \[^3\](14.3% and 17.5%, agent and caller respectively in [4]).

The seed model trained by semi-supervised (1-best) showed 2.0% (agent), 2.3% (caller) absolute WER improvement. The semi-supervised LF-MMI model based on full lattice showed significant improvement from the seed model. Absolute WER improvements were 1.2% for agent channel and 1.3% for caller channel. We could find the fact that WER of agent channel was better than WER of caller channel. This is obvious because our LM contains some portion of agent sentences frequently spoken whereas caller channel (from customer side) sentences are usually hard to predict. Another reason is that quality of agent channel audio is usually better than caller channel which can usually contain background noise (e.g., babble, street, car, etc).

From Table 4, we capture the improvements to the decoding real-time factor and graph, which translates into a significantly smaller production decoding fleet size and lower CPU.

### 4. DISCUSSION

In this report we have chronicled improvements made to our production ASR system based on a semi-supervised training with lattice-based supervision as well as production-scale text selection for language modeling. Our current direction involves experiments to dynamically add words to the FST graph using word-class language models [15, 16]. Other practical solutions involve creating customer and vertical specific training and evaluation sets, including customer-specific LM-rescoring.

While there is a lot of hype and promise around end-to-end (E2E) systems like ESPNET [17, 18] and DeepSpeech [19], our practical experience experimenting and training these models has shown them to be very sensitive to the data cleanliness in ways that encumbers a semi-supervised training process. Further, in comparison with current LF-MMI techniques, E2E performance is not ready for production for the spontaneous, large vocabulary conversational task.

Finally, since the original 30,000 hours of unlabeled audio originally used in [4], we have grown the size of our unsupervised audio dataset to be over 80,000 hours. Future experiments involve using this larger, unfiltered dataset as the starting point for semi-supervised model training efforts.

### 5. ACKNOWLEDGEMENT

The authors thank the following former Marchex colleagues for their efforts and contributions to the Marchex Speech Recognition System: Morten Pedersen and Hari Rajagopal.

### 6. REFERENCES

[1] Wayne Xiong, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. “The Microsoft 2017 conversational speech recognition system,” in 2018 IEEE Interna-
[2] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, Bergul Roomi, and Phil Hall, “English conversational telephone speech recognition by humans and machines,” in Proc. of Interspeech, 2017, pp. 132–136.

[3] Kyu J. Han, Akshay Chandra, Jungsuk Kim, and Ian Lane, “The CAPIO 2017 conversational speech recognition system,” 2017.

[4] Shane Walker, Morten Pedersen, Iroro Orife, and Jason Flaks, “Semi-supervised model training for unbounded conversational speech recognition,” arXiv preprint arXiv:1705.09724, 2017.

[5] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, “Purely sequence-trained neural networks for ASR based on lattice-free MMI,” in Interspeech, 2016, pp. 2751–2755.

[6] George Saon, Hong-Kwang Jeff Kuo, Steven J. Rennie, and Michael Picheny, “The IBM 2015 English conversational telephone speech recognition system,” CoRR, vol. abs/1505.05899, 2015.

[7] George Saon, Tom Sercu, Steven J. Rennie, and Hong-Kwang Jeff Kuo, “The IBM 2016 English conversational telephone speech recognition system,” CoRR, vol. abs/1604.08242, 2016.

[8] Gaurav Kumar, Matt Post, Daniel Povey, and Sanjeev Khudanpur, “Some insights from translating conversational telephone speech,” in Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014, pp. 3231–3235.

[9] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., “The Kaldi speech recognition toolkit,” in IEEE 2011 workshop on automatic speech recognition and understanding, IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[10] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, “A time delay neural network architecture for efficient modeling of long temporal contexts,” in Sixteenth Annual Conference of the International Speech Communication Association, 2015.

[11] Vimal Manohar, Hossein Hadian, Daniel Povey, and Sanjeev Khudanpur, “Semi-supervised training of acoustic models using lattice-free MMI,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 4844–4848.

[12] Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohamadi, and Sanjeev Khudanpur, “Semi-orthogonal low-rank matrix factorization for deep neural networks,” INTERSPEECH (2018, submitted), 2018.

[13] Christopher Cieri, David Miller, and Kevin Walker, “The Fisher corpus: a resource for the next generations of speech-to-text,”.

[14] Kyu J Han, Seongjun Hahm, Byung-Hak Kim, Jungsuk Kim, and Ian Lane, “Deep learning-based telephony speech recognition in the wild,” in Proc. of Interspeech, 2017, pp. 1323–1327.