INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Improved end-of-query detection for streaming speech recognition

Matt Shannon, Gabor Simko, Shuo-yiin Chang, Carolina Parada

Google Inc., USA

{mattshannon,gsimko,shuoyiin,carolinap}@google.com

Abstract

In many streaming speech recognition applications such as voice search it is important to determine quickly and accurately when the user has finished speaking their query. A conventional approach to this task is to declare end-of-query whenever a fixed interval of silence is detected by a voice activity detector (VAD) trained to classify each frame as speech or silence. However silence detection and end-of-query detection are fundamentally different tasks, and the criterion used during VAD training may not be optimal. In particular the conventional approach ignores potential acoustic cues such as filler sounds and past speaking rate which may indicate whether a given pause is temporary or query-final. In this paper we present a simple modification to make the conventional VAD training criterion more closely related to end-of-query detection. A unidirectional long short-term memory architecture allows the system to remember past acoustic events, and the training criterion incentivizes the system to learn to use any acoustic cues relevant to predicting future user intent. We show experimentally that this approach improves latency at a given accuracy by around 100 ms for end-of-query detection for voice search.

Index Terms: endpointing, voice activity detection, end-of-query detection

1. Introduction

In many streaming speech recognition applications such as voice search and dialogue systems it is important to determine quickly and accurately when the user of a system has finished speaking. This task is performed by an *endpointer*, which we term a *microphone closer* or *end-of-query detector* to avoid ambiguity. The system receives a stream of audio and makes a series of binary decisions: to wait for further speech, or to stop listening and submit the audio so far received for subsequent processing. Each of these *mic close* or *stopping* decisions is irrevocable and based only on the audio so far received. It is desirable to have small *latency*, defined as the time between the user finishing speaking and the system closing the mic, and not to *cut off* the user, defined as the system closing the mic before the user has finished speaking. There is a natural tension between these two goals. Mic closer performance can strongly affect users' perceptions of a system. For example mic closer performance is critical to natural turn-taking in dialogue systems and bad mic closer performance has been blamed for low user satisfaction [1–3].

Voice activity detection (VAD), also sometimes known as endpointing, is the task of classifying each frame of audio as either speech (strictly query-related speech) or silence (strictly anything that is not query-related speech). In an offline setting where all the audio is available to the system when making all decisions, VAD and mic closing are effectively the same task, since the end of the last segment of speech is the end of the user's query. However in an online or streaming setting, mic closing is fundamentally harder: a VAD system need only detect any current silence, whereas a mic closer must predict whether there will be subsequent speech.

A simple approach to mic closing is to declare *end-of-query (EOQ)* as soon as a VAD system observes speech followed by a fixed interval, e.g. 300 ms, of silence [4]. Typically the VAD system is obtained by thresholding the posteriors from a probabilistic voice activity classifier. This approach is widely used despite being perceived as unsatisfactory [5, 6]. It seems likely that human listeners use additional acoustic cues such as filler sounds, speaking rhythm or fundamental frequency to inform their view of whether a human talker intends to continue speaking after a given pause [7]. These end-of-query acoustic cues are ignored by VAD-based mic closers.

In this paper we propose using a probabilistic *end-of-query* classifier as the basis for mic closing. The classifier is trained to predict whether or not the user has finished speaking at a given time, and uses a unidirectional LSTM-based architecture to allow its predictions to be informed by past acoustic events. The LSTM and modified loss function are complementary, and the combination offers the potential to automate learning of cues such as filler sounds and past speaking rate which may be temporally isolated from the frames where they are most useful for EOQ prediction and which would be hard to pick up on with simpler models. While conceptually appealing, the modified loss function used in our approach is implementationally straightforward, requiring a simple change to the labels used to train the classifier.

Previous work relevant to improved mic closer performance includes improving the estimate of the duration of current silence [8], and a variety of attempts to go beyond VAD-based mic closing and take end-of-query (sometimes called end-of-utterance) into account [5, 6, 8–10]. Gravano and Hirschberg provide a summary of previous work in this area [7]. Our proposed approach is complementary to this previous work, and may benefit from the EOQ-informative acoustic and decoder features used. Our approach has the advantage of being fully data-driven, and has the potential to extract better EOQ-related information from existing acoustic features by using state-of-the-art sequential models such as LSTMs. There is also a large body of previous work on VAD systems evaluated as VAD systems, but, unfortunately for our purposes, this includes few mic closer–style evaluations.

In the remainder of this paper, we review the conventional approach to training a voice activity classifier and using it for mic closing (§2), describe our proposed approach to training a end-of-query classifier and using that for mic closing (§3), discuss metrics relevant to evaluating a mic closer (§4), and present our experimental results (§5).

http://dx.doi.org/10.21437/Interspeech.2017-496

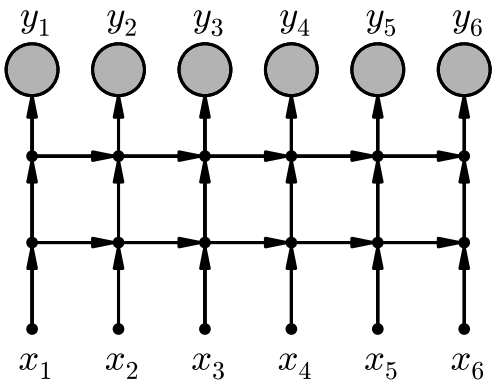

Figure 1: *Directed graphical model showing the computational and conditional independence structure of the LSTM-based classifier considered in this paper. Solid nodes are deterministic given their parents whereas circular nodes are stochastic, and observed nodes are shaded. Here x_t is an acoustic feature vector and y_t is a binary label. For the traditional voice activity classifier y_t is* speech *or* silence*, whereas for the proposed end-of-query classifier y_t is* query-not-complete *or* query-complete*.*

2. Voice activity classifier–based mic closing

In this section we describe a widely used approach to mic closing based on training an online or streaming probabilistic voice activity classifier.

A conditional probabilistic model $\mathbb{P}(y|x,\lambda)$ used during training specifies the probability of a sequence $y = [y_t]_{t=1}^T$ of speech / silence labels given an acoustic feature vector sequence $x = [x_t]_{t=1}^T$ and model parameters λ. For simplicity we typically assume the labels $y_1, y_2, \ldots$ at different times are conditionally independent, even though this is unlikely to be true. The probability $\mathbb{P}(y_t|x,\lambda)$, often called the "posterior", is given by the output of a neural net which takes the acoustic feature vector sequence as input. We use a recurrent architecture including one or more *long short-term memory (LSTM)* [11] layers to allow the system to remember past acoustic information relevant to predicting whether the current frame is speech or silence. The recurrent layers are unidirectional to allow the overall system to operate in a streaming fashion. The final layer is a 2-class softmax layer which outputs framewise speech and silence posteriors. A directed graphical model showing the model structure is shown in Figure 1. The probabilistic model is trained using maximum likelihood (i.e. cross-entropy). The reference speech / silence label sequence used for training may be obtained by forced alignment of a human reference transcript, labeling all non-silence phonemes as speech. For concreteness we use 1 for a speech label and 0 for silence.

To use the trained probabilistic voice activity classifier for mic closing, the framewise posteriors are thresholded to obtain hard speech / silence decisions, and the mic is closed as soon as the system observes some speech followed by a fixed time interval of silence.

The above training procedure incentivizes the system to detect the acoustic cues which distinguish present speech from present silence, but ignores cues which may help to predict whether a current silence will be followed by subsequent speech.

3. End-of-query classifier–based mic closing

In this section we describe our proposed approach to mic closing based on training a probabilistic *end-of-query classifier* to directly predict whether or not the user has finished speaking at a given time.

The probabilistic model $\mathbb{P}(y|x,\lambda)$ has the same structure described in §2 but uses different labels during training; the labels are now query-not-complete (label 1) or query-complete (label 0). The reference label sequence used during training always consists of a sequence of 1s followed by a sequence of 0s, with the first 0 occurring at the time of the ideal mic close. The ideal mic close occurs immediately after the user has finished speaking. An example of these *VAD-style* and *EOQ-style* label sequences is shown in Table 1.

Table 1: *Example of the difference between VAD-style (silence is* 0 *and speech is* 1*) and EOQ-style (query-complete is* 0 *and query-not-complete is* 1*) targets used during classifier training for an utterance with* 10 *frames where the user finishes speaking at frame* 8*.*

frame	0	1	2	3	4	5	6	7	8	9
VAD-style	0	0	1	1	1	0	1	1	0	0
EOQ-style	1	1	1	1	1	1	1	1	0	0

To use the trained probabilistic end-of-query classifier for mic closing, the framewise posteriors are thresholded to obtain hard end-of-query decisions, and the mic is closed as soon as the system first outputs a query-complete label 0. The hard thresholding is a heuristic procedure and likely suboptimal in terms of "maximizing utility", but is simple and effective.

This straightforward change in training data incentivizes the system to detect any acoustic cues which help indicate whether the user intends to utter more speech. For example, if a user says "um" during a longish pause, the end-of-query classifier has the power (due to the LSTM) and inclination (due to the modified loss function) to remember that acoustic event and decrease the probability of query-complete in subsequent silence frames.

The posteriors for a sample utterance are shown in Figure 2. It can be seen that the belief of the end-of-query classifier in query-complete grows during periods of non-initial silence, but that the rate is not linear: in the first pause shown the system is relatively uncertain of end-of-utterance and the posterior grows slowly, for example. The difference the training criterion makes can also be seen in the fact that the voice activity classifier treats the silences near the start and end of the utterance in the same way, whereas the end-of-query classifier treats them very differently.

4. Mic closer metrics

After much experimentation, we have settled on a set of four metrics which we feel together give good insight into mic closer performance, and which we use at Google when deciding whether to launch a new system.

The metrics are summarized in Table 2. Word error rate is the primary metric of speech recognition accuracy and so widely perceived as important. It is strongly affected by the mic closer since a cutoff often cuts off many words. EP cutoff is the proportion of utterances where the user is cut off, i.e. the system closes the mic before the user has finished speaking their query. This is an important quantity to measure since being cut

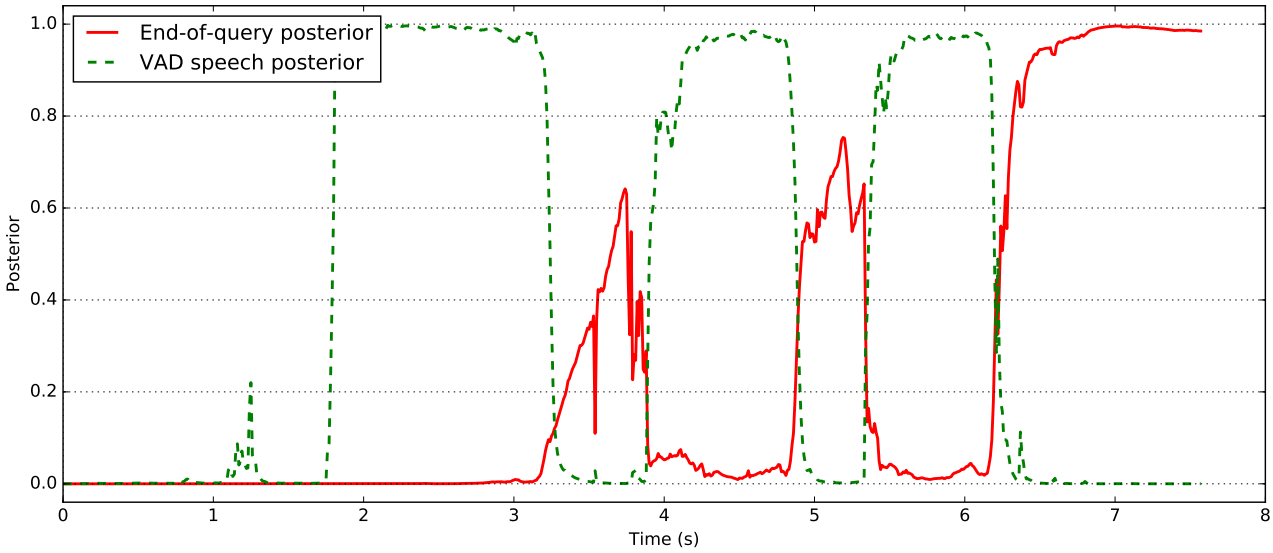

Figure 2: *Posteriors from our trained voice activity classifier and end-of-query classifier for a single utterance.*

off is a negative user experience and, in applications such as voice search, often requires repeating the entire query. Together WER and EP cutoff measure the *accuracy* of a mic closer. EP50 is the median latency over all utterances. This provides an idea of the typical user experience when using the system. EP90 is the 90^{th} percentile latency over all utterances. This tail latency provides an idea of how bad the user experience is when the system is slow to close the mic. Here latency is the time at which the system closed the mic minus the time at which the user finished speaking, and is negative for utterances where the system cut off the user. Together EP50 and EP90 measure the *speed* of a mic closer. For EP cutoff, EP50 and EP90, forced alignment of a reference transcript is used to determine when the user finished speaking.

Table 2: *Metrics used to evaluate mic closer performance.*

WER	Word error rate
EP cutoff	Proportion of utterances with negative latency
EP50	Median latency over all utterances
EP90	90^{th} percentile latency over all utterances

Note that the traditional metrics used to evaluate a voice activity detector such as false alarm rate and false reject rate or precision and recall are not very useful for evaluating the performance of a mic closer.

5. Experiments

We evaluated the performance of the conventional voice activity classifier–based mic closer presented in §2 and the end-of-query classifier–based mic closer proposed in §3 for voice search.

The input acoustic feature vector sequence consisted of 40-dimensional log mel filterbanks with an upper limit of 4 kHz and a frame step of 10 ms using a 25 ms window. This is passed into a frequency convolutional layer with a filter width of 8 frequency bins and pooling with stride 3, followed by a 64-node ReLU DNN layer, a sequence of two 64-cell LSTM layers, another 64-node ReLU DNN layer, and a 2-node softmax layer. The combination of convolutional, stacked LSTM and DNN layers is referred to as a CLDNN [12]. LSTM-based architectures, and in particular CLDNNs, have previously been shown to work well for VAD [13, 14]. In preliminary experiments we observed small but consistent gains from using a CLDNN rather than a stacked LSTM for the end-of-query classifier.

The training data was a voice search–specific subset of a corpus of around 22 million anonymized utterances. Forced alignment to determine ground truth speech and silence regions was performed using a separately trained DNN-based recognizer. Our evaluation set is a collection of 15 thousand anonymized voice search utterances where the production mic closer was intentionally delayed to allow better evaluation of the case where a proposed system closes later than production.

We trained using asynchronous stochastic gradient descent with a fixed learning rate of 2×10^{-5} for around 100 million steps, where each step computed the gradient on a single whole utterance. We selected the model to evaluate based on frame accuracy on a much smaller held-out set taken from the same distribution as the training set, and the selected system was between 50 and 100 million steps in all cases.

We evaluated the three systems shown in Table 3 as mic closers. The VAD+state system post-processes the hard speech / silence decisions produced by the conventional VAD system and is designed to smooth its output and remove erroneous transient speech and silence segments. From a mic closing perspective this has the consequence that we wait not for, say, the past 30 frames to be classified as silence but rather for, say, 90% of the past 40 frames to be classified as silence.

Table 3: *Mic closers compared in our experiments.*

VAD	Thresholded voice activity classifier
VAD+state	VAD post-processed with a state machine
EOQ	Thresholded end-of-query classifier

The relationship between word error rate (WER), median latency (EP50) and tail latency (EP90) is shown in Figure 3 and Figure 4. Each curve was generated by sweeping the threshold

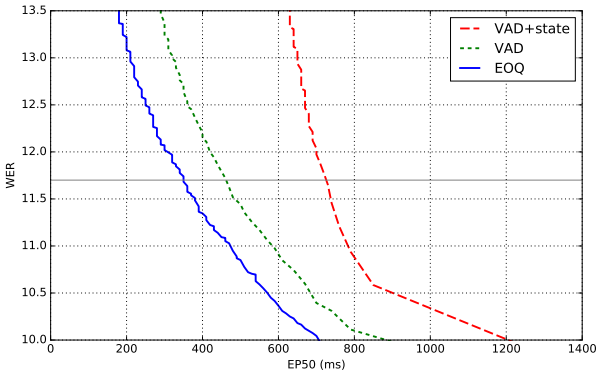

Figure 3: *WER vs EP50 trade-off for the models. The horizontal gray line shows our preferred operating point of WER 11.7.*

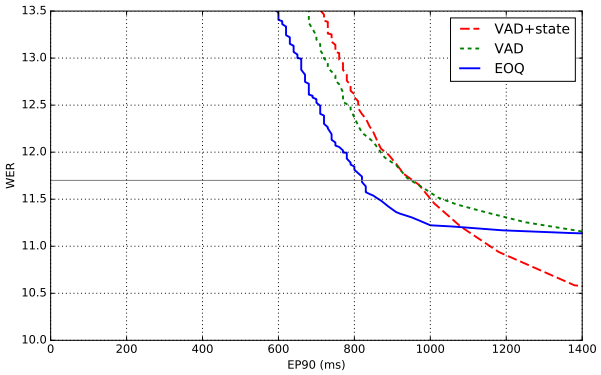

Figure 4: *WER vs EP90 trade-off for the models. The horizontal gray line shows our preferred operating point of WER 11.7.*

used to convert the posteriors into hard decisions. The trade-off between speed and accuracy inherent to the mic closing task can clearly be seen: around our preferred operating point of a WER of 11.7, each 0.2 improvement in absolute WER comes at the expense of around 30 ms in EP50 or 30 to 60 ms in EP90. The EOQ system clearly has the best performance, followed by the VAD system. The VAD+state system performs better than the other systems at high latencies (not shown). However around our preferred operating point it introduces too much latency to be competitive as a mic closer. Performance at a matched word error rate of 11.7 is shown in Table 4. The EOQ system reduces typical latency by 110 ms and tail latency by 120 ms compared to the VAD system.

Table 4: *Performance of mic closers based on VAD and EOQ classifiers.*

Classifier	WER	EP cutoff	EP50 / ms	EP90 / ms
VAD+state	11.7	4.8%	730	970
VAD	11.7	4.9%	460	940
EOQ	11.7	5.1%	350	820

The thresholded VAD classifier has two parameters: threshold value and amount of time to wait before declaring end-of-query. The threshold EOQ classifier has just the threshold value, though we also experimented with set-ups where the system waited for a certain interval of query-complete outputs before declaring end-of-query. In the experiments for this paper we swept over possible *wait time* values of 0, 100, 200 and 300 ms for all systems. For all three systems we found 0 ms to be the best around the operating point of interest, and so all figures reported above use this value. It is somewhat intuitive that this might be a reasonable value for the EOQ system and the VAD+state system, since the state machine increases both latency and confidence in predictions. However it is very surprising for the VAD system; this means it leads to better mic closer performance to wait for 0 ms of extremely high confidence silence (threshold extremely close to 0) than to wait for, say, 200 ms of fairly high confidence silence. The difference between the 0 ms VAD system and the 100 ms and 200 ms systems in terms of WER vs EP50 was not large though. For the experiments in this paper we trained on relatively clean data. In set-ups where we train with noisified data the optimal mic closer performance for the VAD system is indeed obtained using a larger wait time interval such as 200 ms.

We also experimented with using an "output delay" K so that the VAD classifier has seen frames up to frame $t+K$ when trying to predict the label y_t. This introduces latency but may improve the robustness of the classifier by making its decisions based on more information. However in preliminary experiments using an output delay of $K = 5$ was not found to give an improvement in performance of the VAD as a mic closer.

6. Conclusion

We have shown that using an end-of-query classifier rather than the conventional voice activity classifier for mic closing improves both typical and tail latency by around 100 ms at a given accuracy. Training an end-of-query classifier requires a simple tweak to the procedure used to train a conventional voice activity classifier. We have also discussed metrics relevant to evaluating mic closers.

7. Future work

We plan to conduct an analysis of utterances where the EOQ system closes the mic at around the right time while the VAD system closes much too early, to investigate whether it is easy to interpret the type of acoustic cues the EOQ system is using.

8. References

[1] R. Porzel and M. Baudis, "The Tao of CHI: Towards Effective Human-Computer Interaction," in *Proc. HLT-NAACL*, 2004.

[2] N. G. Ward, A. G. Rivera, K. Ward, and D. G. Novick, "Root causes of lost time and user stress in a simple dialog system," in *Proc. Interspeech*, 2005.

[3] A. Raux, D. Bohus, B. Langner, A. W. Black, and M. Eskenazi, "Doing research on a deployed spoken dialogue system: one year of let's go! experience." in *Proc. Interspeech*, 2006.

[4] R. Hariharan, J. Hakkinen, and K. Laurila, "Robust end-of-utterance detection for real-time speech recognition applications," in *Proc. ICASSP*, 2001.

[5] L. Ferrer, E. Shriberg, and A. Stolcke, "Is the speaker done yet? Faster and more accurate end-of-utterance detection using prosody," in *Proc. ICSLP*, 2002.

[6] ——, "A prosody-based approach to end-of-utterance detection that does not require speech recognition," in *Proc. ICASSP*, 2003.

[7] A. Gravano and J. Hirschberg, "Turn-taking cues in task-oriented dialogue," *Computer Speech & Language*, vol. 25, no. 3, pp. 601–634, 2011.

[8] B. Liu, B. Hoffmeister, and A. Rastrow, "Accurate endpointing with expected pause duration," in *Proc. Interspeech*, 2015.

[9] A. Raux and M. Eskenazi, "Optimizing endpointing thresholds using dialogue features in a spoken dialogue system," in *Proc. SIGdial*, 2008.

[10] D. Schlangen, "From reaction to prediction: Experiments with computational models of turn-taking," *Proc. Interspeech*, 2006.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. ICASSP*, 2015.

[13] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with LSTM recurrent neural networks and an application to Hollywood movies," in *Proc. ICASSP*, 2013.

[14] R. Zazo Candil, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform CLDNNs for voice activity detection," in *Proc. Interspeech*, 2016.