Query-by-Example Search with Discriminative Neural Acoustic Word Embeddings

Shane Settle[1], Keith Levin[2], Herman Kamper[1], Karen Livescu[1]

[1]TTI-Chicago, USA [2]Johns Hopkins University, USA

Introduction

- Query-by-example (QbE) speech search is the task of searching for a spoken query term in a collection of speech recordings
- This task arises naturally when the search terms may be out-of-vocabulary, in hands-free settings, or in low-resource settings
- Prior work largely based on DTW (e.g. [1])
- Some recent work has explored using fixed-dimensional embeddings to represent both query and database segments, and nearest-neighbor search to determine putative matches(e.g. [2])
- This work: A neural embedding model for representing query and database segments, learned from limited labeled data using a contrastive loss
- We improve over past techniques which rely on DTW [1] and template-embedding [2] based methods for segment comparison

Experimental Setup

Training the Neural Acoustic Word Embedding (NAWE) model [3, 4, 5]:

- Standard train(10k)/dev(11k) partitioning of Switchboard conversational corpus from prior work (dev used to tune based on the word discrimination task [6])
- Acoustic features used are 39-dimensional MFCCS+Δ+ΔΔs

Evaluating on the QbE task [2, 7]:

- 37-hour set from which query terms are drawn
- 48-hour development search collection to tune hyperparameters
- 433-hour evaluation search collection.

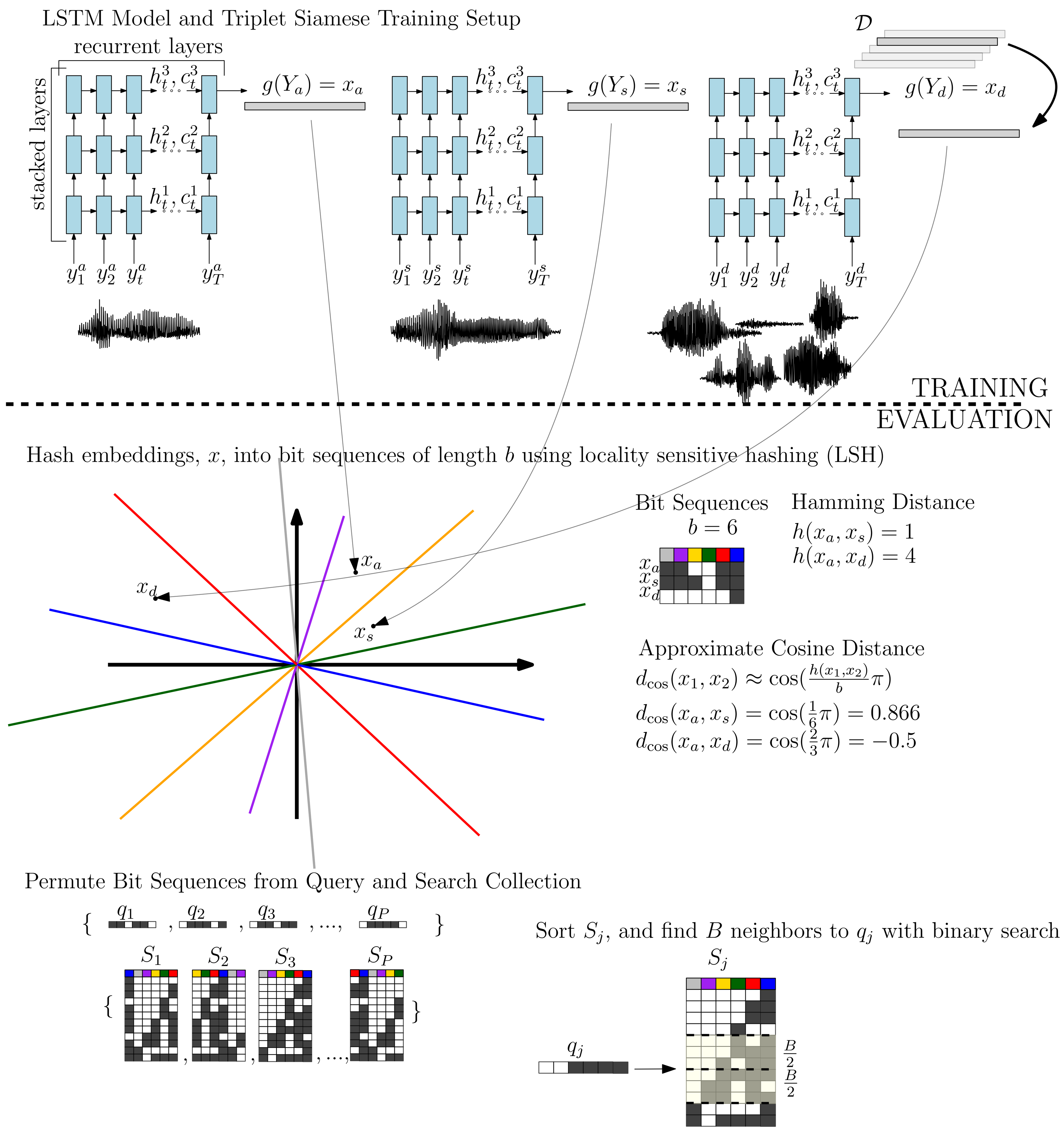
Evaluation metrics:

- *figure-of-merit* (FOM): recall averaged over the ten operating points at which the false alarm rate per hour of search audio is equal to $1, 2, \ldots, 10$
- *oracular term weighted value* (OTWV): query-specific weighted difference between recall and false alarm rate
- *precision at 10* (P@10): the fraction of the top ten results which are correct matches to the query

NAWE-based QbE

NAWE Training Objective

$$l_{\text{cos hinge}} = \max\{0, m + d_{\cos}(x_a, x_s) - \max_{x_d \in \mathcal{D}} d_{\cos}(x_a, x_d)\}$$



Evaluation Results

System	Median Example			Best Example			Query Time (s)
	FOM	OTWV	P@10	FOM	OTWV	P@10	
DTW-based [1]	6.7	2.7	44.0	20.7	10.4	84.4	24.7
Template-based [2]	24.5	14.4	34.5	46.2	26.6	87.4	0.078
Ours	43.3	22.4	60.2	65.4	43.3	95.1	0.38

Table: Comparison of QbE system performance on the evaluation set. Hyperparameters are set to $b = 1024$, $P = 16$, $B = 2,000$.

Running time vs Precision@10

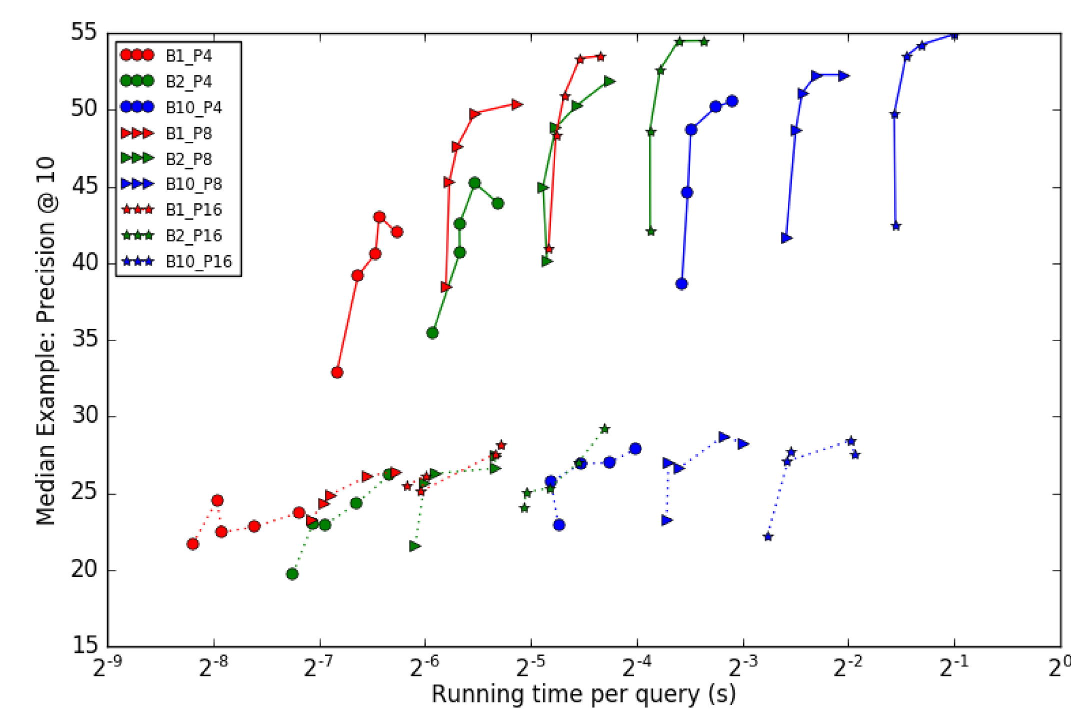


Figure: Ours (solid) vs. S-RAILS [2] (dotted) on the development search collection; connected points indicate a system with fixed permutation number (P) and beamwidth (B) while signature length (b) is varied from 128 to 2048.

Development Results and Observations

	Median Example			Best Example		
	FOM	OTWV	P@10	FOM	OTWV	P@10
vary b						
128	62.1	37.4	42.1	81.7	60.8	83.8
256	67.2	42.6	48.6	83.0	65.4	84.9
512	68.2	44.8	52.6	83.6	65.9	84.9
1024	69.1	46.5	54.5	84.1	66.7	84.8
2048	**70.4**	**48.3**	**54.5**	**85.0**	**66.8**	**86.0**
vary P						
4	48.8	33.2	45.2	75.2	59.0	83.0
8	60.9	41.0	50.3	80.3	63.8	**85.0**
16	**69.1**	**46.5**	**54.5**	**84.1**	**66.7**	84.8
vary B						
1000	65.8	44.8	53.4	83.0	65.6	**85.0**
2000	69.1	46.5	**54.5**	84.1	66.7	84.8
10000	**74.6**	**49.5**	54.2	**86.3**	**67.9**	84.8

Table: Effect of varying signature length b, number of permutations P, and beamwidth B on dev performance; when fixed, parameters are set to $b = 1024$, $P = 16$, $B = 2,000$.

- Increases in signature length and # of permutations yield larger improvements in P@10 for our system than the template-embedding system (S-RAILS)
- Performance benefits from increasing signature length and number of permutations saturate later for our system
- When varying P, performance has not plateaued for Median Example, continued exploration in this direction may further improve results
- Increasing beamwidth does not efficiently increase P@10 performance, but helps to improve recall, as seen in the FOM score

Queries and Top-Hits

Figure: Embeddings of queries and their top hits, visualized using t-SNE. Queries are shown in large capital letters, while the top several hits for each query is shown in the same color as the query. Random segments from the search collection and their associated transcriptions are shown in gray.

Conclusion

- NAWEs give relative improvement over template-embeddings (S-RAILS) of >55% across all metrics for Median Example results
- Directions for future work:
  - Explore limits of the approach as the amount of training data is varied
  - Train a QbE system end-to-end
  - Joint models for both QbE and text-based spoken term detection

References

[1] A. Jansen and B. Van Durme, "Indexing raw acoustic features for scalable zero resource search." in *Interspeech*, 2012, pp. 2466–2469.

[2] K. Levin, A. Jansen, and B. Van Durme, "Segmental acoustic indexing for zero resource keyword search," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc.*, 2015.

[3] H. Kamper, M. Elsner, A. Jansen, and S. J. Goldwater, "Unsupervised neural network based feature extraction using weak top-down constraints," in *Proc. ICASSP*, 2015.

[4] S. Settle and K. Livescu, "Discriminative acoustic word embeddings: Recurrent neural network-based approaches," in *Proc. IEEE Workshop on Spoken Language Technology (SLT)*, 2016.

[5] W. He, W. Wang, and K. Livescu, "Multi-view recurrent neural acoustic word embeddings," in *Proc. Int. Conf. Learning Representations*, 2017.

[6] M. A. Carlin, S. Thomas, A. Jansen, and H. Hermansky, "Rapid evaluation of speech representations for spoken term discovery," in *Proc. Interspeech*, 2011.

[7] A. Jansen and B. Van Durme, "Indexing raw acoustic features for scalable zero resource search," in *Proc. Interspeech*, 2012.