Google

Parameter-Free Attentive Scoring for Speaker Verification

Jason Pelecanos, Quan Wang,
Yiling Huang, Ignacio Lopez Moreno
[paper]

ODYSSEY
June 28th-July 1st 2022, Beijing, China

Agenda

Section 1

Background

Embedding based similarity scoring methods

- Common approaches
  - Cosine scoring
  - Softmax Cross Entropy training followed by a PLDA stage [Snyder, 2018]
  - Added Margins (for example, ECAPA-TDNN system [Desplanques, 2020])
- Uncertainty Statistics
  - Neural network estimates embeddings which include uncertainty statistics for parametric scoring [Silnova, 2020]
- Decision-Residual Vectors (Dr-Vectors)
  - Compare embeddings using a small neural network as a residual estimate [Pelecanos, 2021]
- Attention based neural network scoring
  - Learns an attention based neural network [Li, 2020; Jung, 2021]

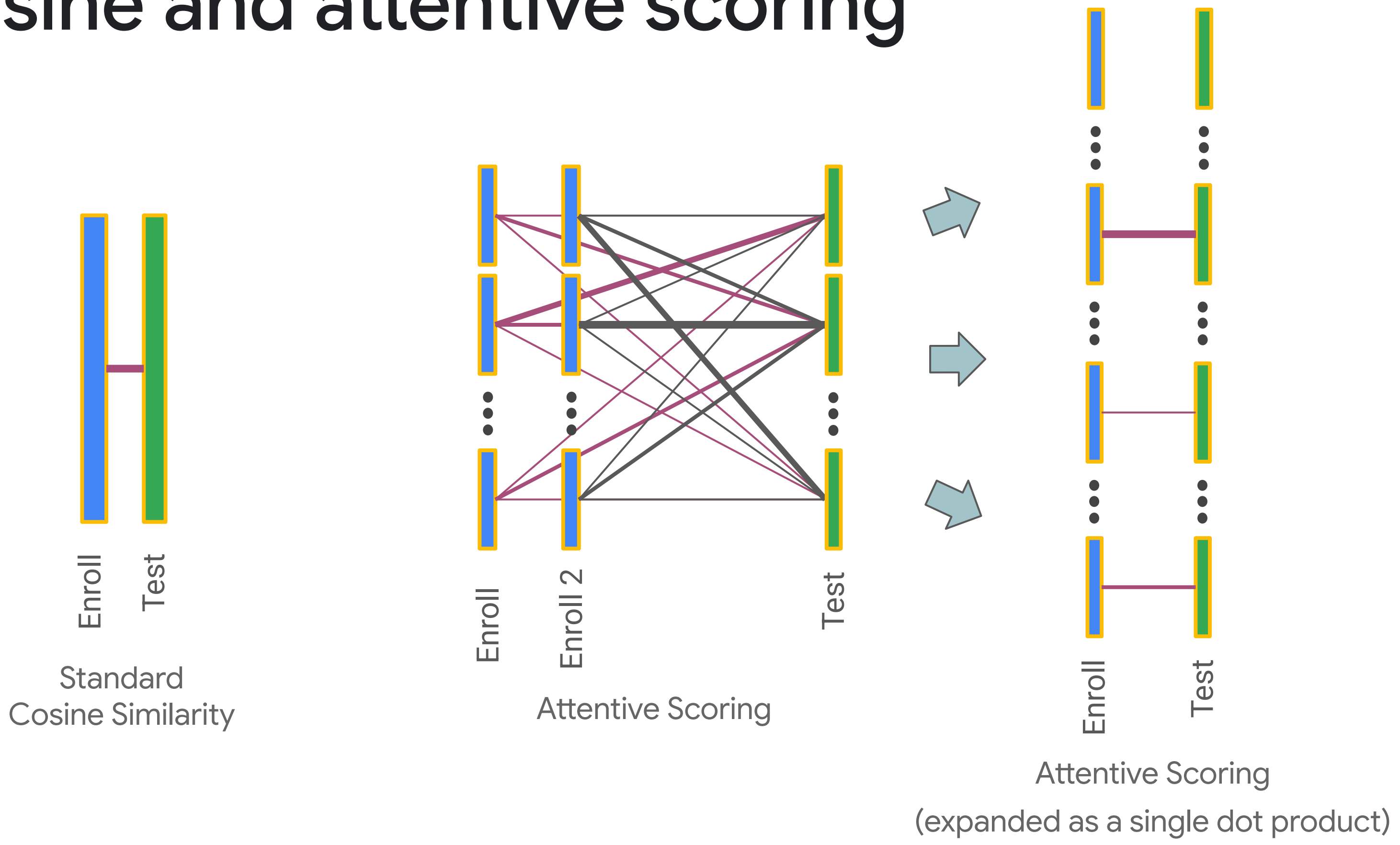
Attentive scoring: A simple approach?

- We currently use cosine based similarity scoring for comparing speakers
  - For multiple enrollment utterances, embeddings are averaged.
  - Not ideal if enrollment utterances are from different and perhaps more relevant devices
- How can we improve performance for multiple enrollment utterances?
  - “Attend” (weight) the enrollment utterances depending on relevance to the test utterance.
  - We can also attend to relevant parts within an utterance (not the initial focus).
- How can we use attention for scoring?
  - We can adapt the scaled dot-product attention from the Transformers paper [Vaswani, 2017] to the speaker recognition problem.

Section 2

Parameter-Free Attentive Scoring

Cosine and attentive scoring



Parameter-free attentive scoring

Cosine Similarity:

$$s_{\cos}(\mathbf{t}, \mathbf{e}) = \frac{\mathbf{t} \cdot \mathbf{e}}{\|\mathbf{t}\|_2 \|\mathbf{e}\|_2}$$

queries values

Test ⇨ $\mathbf{U}_t = \{\mathbf{q}_m, \mathbf{t}_m\}_{m=1,\ldots,M}$

Enroll ⇨ $\mathbf{U}_e = \{\mathbf{k}_n, \mathbf{e}_n\}_{n=1,\ldots,N}$

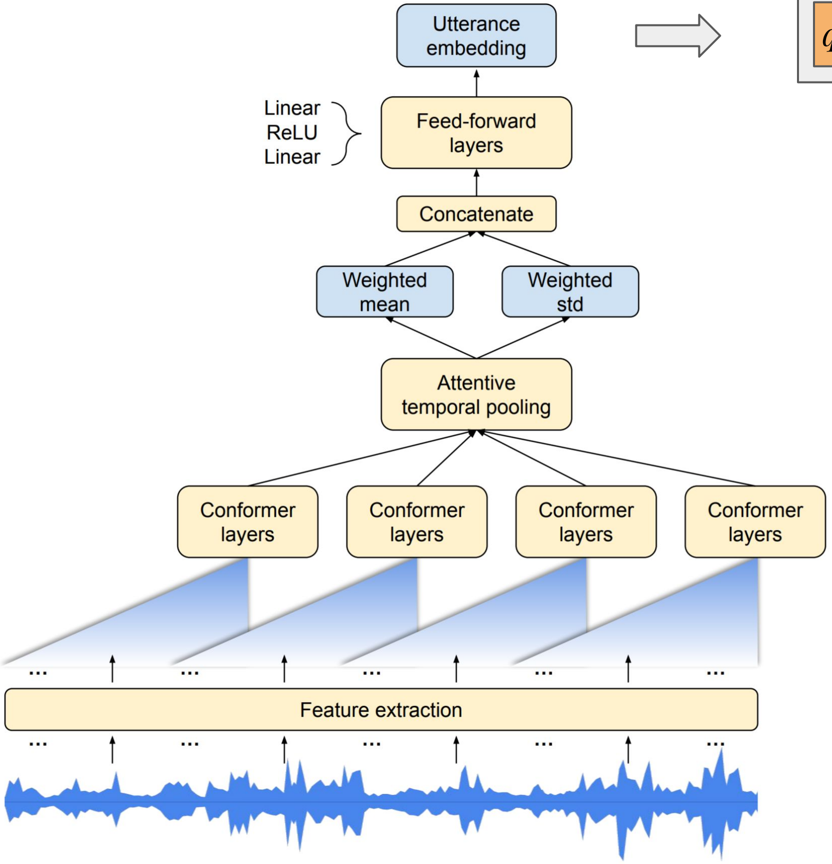
keys values

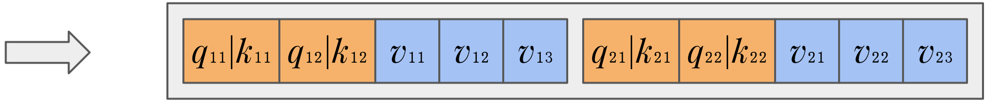
$$s_{\text{att}}(\mathbf{U}_t, \mathbf{U}_e) = \sum_{m=1}^{M} \sum_{n=1}^{N} w_{mn} \mathbf{t}_m \cdot \mathbf{e}_n$$

learned scale

$$w_{mn} = \frac{\exp(\alpha\, \mathbf{q}_m \cdot \mathbf{k}_n)}{\sum_{i=1}^{M} \sum_{j=1}^{N} \exp(\alpha\, \mathbf{q}_i \cdot \mathbf{k}_j)}$$

Model architecture





Section 3

Results

Data

- Speech data
    - Vendor collected speech data, LibriVox, CN-Celeb & LDC sourced data.
    - Most recordings are short utterance recordings of a few seconds using different devices such as cell phone or laptop
    - Training and evaluation data are also augmented with noisy speech based on simulated additive noise and reverberation effects
- Training
    - 9 languages (+ other supplementary languages)
    - 230k speakers, 74M utterances
- Evaluation:
    - 9 languages
    - 25k speakers, 1.7M utterances
    - Trials per language: 93-200k target trials, 200k non-target trials

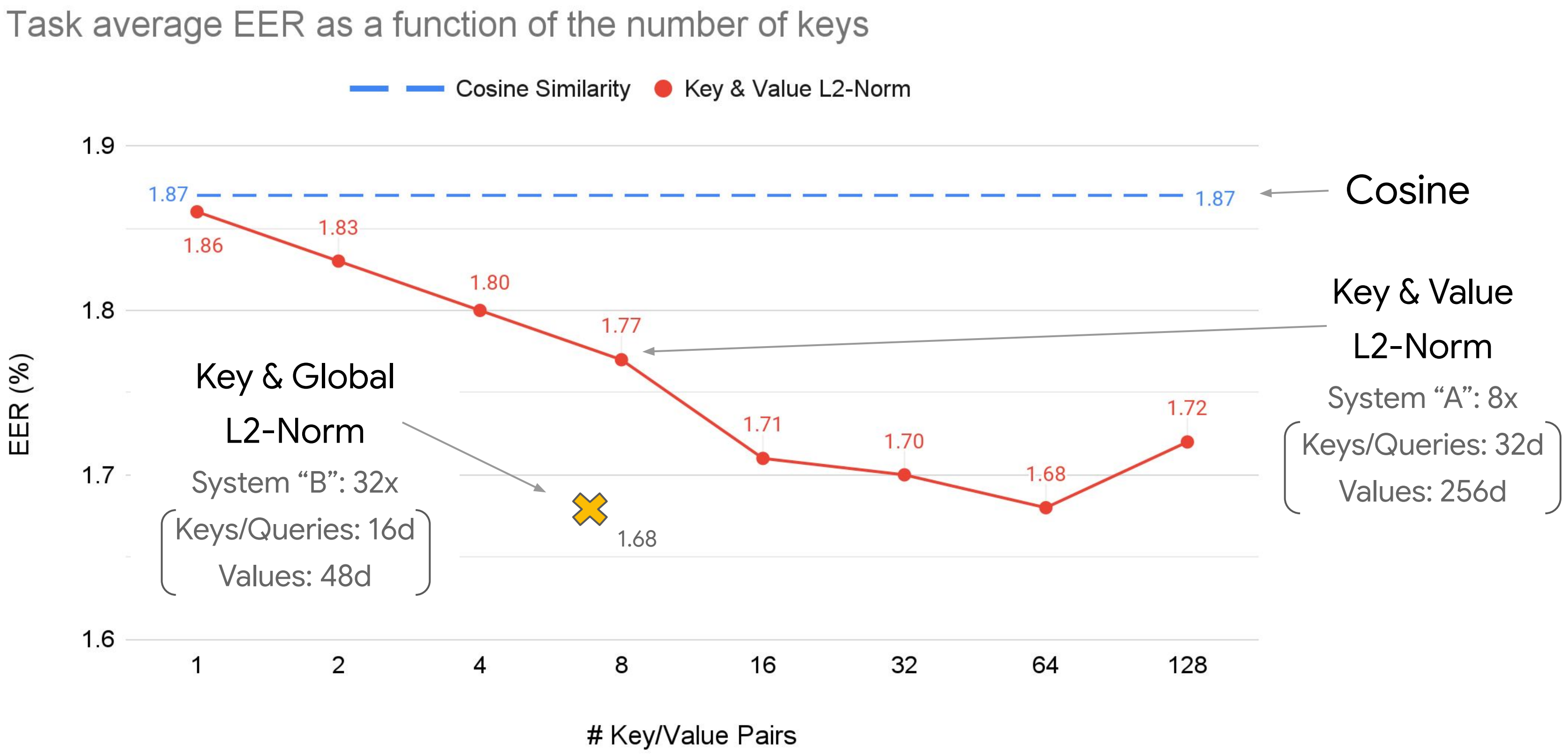
Results overview

- Comparison across normalizations
- Varying the number of keys
- Tied versus Independent keys and queries
- Mean versus joint estimation for embeddings

Comparison across normalizations

System	# Key-Value Pairs	# Dimensions Per Key	Per Value	Total	Single Enroll EER (%) Clean	Noisy	Multi Enroll EER (%) Clean	Noisy	Task Average EER (%)
Baseline (Cosine Similarity)	-	-	-	256	2.23	3.38	0.67	1.23	1.88
	-	-	-	512	2.31	3.34	0.68	1.20	1.88
	-	-	-	2304	2.22	3.35	0.67	1.23	1.87
Attentive Scoring:									
No normalization	8	32	256	2304	2.15	3.26	0.69	1.29	1.85
Layer Normalization [19]	8	32	224	2048	2.15	3.30	0.71	1.30	1.86
Key & Value L2-Norm (A)	8	32	256	2304	2.03	3.20	0.65	1.21	1.77
Key & Value L2-Norm	32	16	48	2048	2.01	3.20	0.65	1.22	1.77
Key & Global L2-Norm	8	32	256	2304	1.94	3.15	0.61	1.20	1.72
Key & Global L2-Norm (B)	32	16	48	2048	**1.93**	**3.02**	**0.60**	**1.15**	**1.68**
			Relative Improvement:		13%	10%	10%	4%	10%

Varying the number of keys



Section 4

Conclusions

Conclusions

- Proposed a parameter-free attentive scoring approach.
  - Relatively straightforward to implement
  - Scoring is a simple function of the embeddings (+ scaling factor)
- Results
  - Attentive scoring **normalization** plays an important role in system performance.
    - Overall "Global L2-normalization" performed best
  - Significantly increasing the number of keys can help (at the cost of more parameters).
  - Also explored the effects of:
    - Tied/Independently estimated keys/queries
    - Mean/Joint estimation for multiple enrollment utterances
- Future work
  - Significant scope for exploring other configurations/normalizations.

Additional Material

Global L2-normalization

- Similarity represented as one large dot-product:

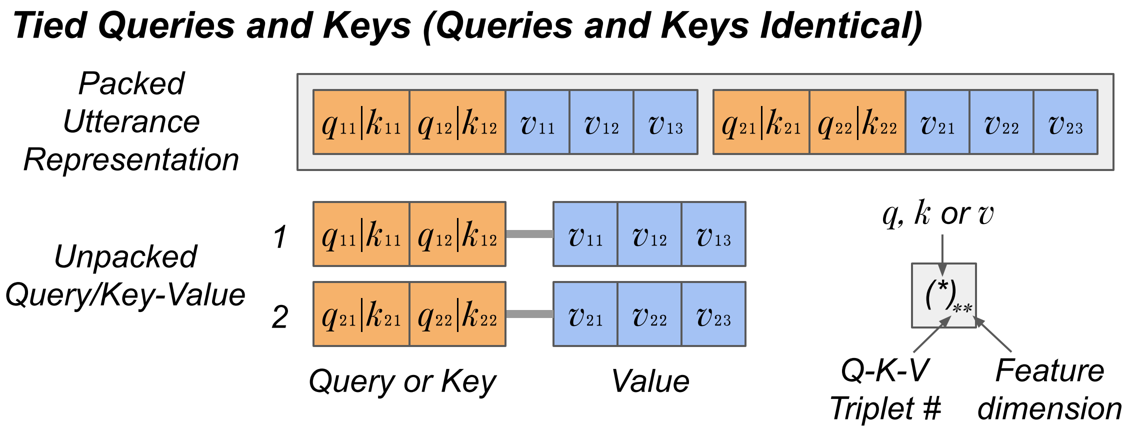
$$s_{\text{att}}(\mathbf{U}_t, \mathbf{U}_e) = \mathbf{a} \cdot \mathbf{b}$$

$$\mathbf{a} = \begin{pmatrix} \begin{pmatrix} \sqrt{w_{11}}\mathbf{t}_1 \\ \vdots \\ \sqrt{w_{1N}}\mathbf{t}_1 \end{pmatrix} \\ \vdots \\ \begin{pmatrix} \sqrt{w_{M1}}\mathbf{t}_M \\ \vdots \\ \sqrt{w_{MN}}\mathbf{t}_M \end{pmatrix} \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} \begin{pmatrix} \sqrt{w_{11}}\mathbf{e}_1 \\ \vdots \\ \sqrt{w_{1N}}\mathbf{e}_N \end{pmatrix} \\ \vdots \\ \begin{pmatrix} \sqrt{w_{M1}}\mathbf{e}_1 \\ \vdots \\ \sqrt{w_{MN}}\mathbf{e}_N \end{pmatrix} \end{pmatrix}$$

- Calculate cosine similarity:

$$s_{\|\text{att}\|}(\mathbf{U}_t, \mathbf{U}_e) = \frac{s_{\text{att}}(\mathbf{U}_t, \mathbf{U}_e)}{\sqrt{\|\mathbf{a}\|_2^2 \ \|\mathbf{b}\|_2^2}} = \frac{\mathbf{a} \cdot \mathbf{b}}{\sqrt{\|\mathbf{a}\|_2^2 \ \|\mathbf{b}\|_2^2}}$$

How do we generate queries, keys & values?



- Similar packing/unpacking can be achieved for independent query and key estimation

Data

Language	Training Spk [k]	Training Utt [k]	Evaluation Spk [k]	Evaluation Utt [k]	Evaluation Tar [k]	Evaluation Non [k]
English (India)	6.0	2900	1.6	264	200	200
English (US)	46.7	3329	12.4	543	200	200
French	5.8	2458	1.4	161	152	200
Hindi	8.3	1642	2.4	106	93	200
Italian	5.2	2251	1.2	102	95	200
Japanese	6.2	2048	1.7	106	97	200
Korean	5.4	1407	1.5	160	151	200
Portuguese (Brazil)	5.8	1675	1.5	107	97	200
Spanish	4.6	2189	1.2	113	106	200
Other Data/Sources†	139.8	54499	-	-	-	-

† *Includes vendor collected data from languages outside of the languages mentioned as well as LibriVox, CN-Celeb, and LDC sourced data.*

Tied and independently estimated keys/queries

System	Query & Key	Total Dims	Single Enroll EER (%) Clean	Single Enroll EER (%) Noisy	Multi Enroll EER (%) Clean	Multi Enroll EER (%) Noisy	Task Average EER (%)
(A)	Tied	2304	2.03	3.20	0.65	1.21	1.77
	Indep	2560	2.06	3.08	0.63	**1.14**	1.73
(B)	Tied	2048	**1.93**	**3.02**	**0.60**	1.15	**1.68**
	Indep	2560	2.04	3.12	0.62	1.17	1.74

System “A”: 8x[Keys/Queries: 32d, Values: 256d]

System “B”: 32x[Keys/Queries: 16d, Values: 48d]

Varying the number of keys

# Key-Value Pairs	Total Dims	Single Enroll EER (%)		Multi Enroll EER (%)		Task Average EER (%)
		Clean	Noisy	Clean	Noisy	
1	288	2.17	3.20	0.75	1.31	1.86
2	576	2.08	3.23	0.70	1.30	1.83
4	1152	2.12	3.21	0.68	1.21	1.80
8	2304	2.03	3.20	0.65	1.21	1.77
16	4608	1.99	**3.09**	0.62	**1.15**	1.71
32	9216	1.95	3.10	0.61	1.16	1.70
64	18432	**1.89**	3.10	**0.60**	**1.15**	**1.68**
128	36864	1.97	3.10	0.61	1.18	1.72

System “A”: ?x[Keys/Queries: 32d, Values: 256d]

Mean versus joint estimation

Estimation Method Train	Estimation Method Eval	Single Enroll EER (%) Clean	Single Enroll EER (%) Noisy	Multi Enroll EER (%) Clean	Multi Enroll EER (%) Noisy	Task Average EER (%)
Joint	Joint	**2.03**	**3.20**	**0.65**	1.21	1.77
Joint	Mean	**2.03**	**3.20**	**0.65**	1.16	**1.76**
Mean	Mean	2.32	3.57	0.67	**1.14**	1.92

System “A”: 8x[Keys/Queries: 32d, Values: 256d]