



Supervised Deep Hashing for Efficient Audio Retrieval

Arindam Jati

Audio & Acoustics Research Group, led by Ivan Tashev

Mentor: Dimitra Emmanouilidou

Microsoft Research - Redmond

Intro

- University of Southern California (USC), Los Angeles
- Signal Analysis and Interpretation Laboratory (SAIL)
- Advisor: **Prof. Shrikanth Narayanan**
- <https://sail.usc.edu/>



Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- Efficient audio retrieval with hashing
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- Conclusions and future work

Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- Efficient audio retrieval with hashing
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- Conclusions and future work

Audio Event Detection & Classification

Definition, Human annotation

- “Human-like ability to identify and relate sounds from audio” (Gemmeke, et al.)
- Audio event annotations:
 - Human annotators
 - Provide semantic label to a sound
 - Generally follow an ontology or hierarchy during annotations
 - e.g. Google AudioSet ontology (Gemmeke, et al.)

○ Human sounds

- *Human voice*
- *Whistling*
- *Respiratory sounds*
- *Human locomotion*
- *Digestive*
- *Hands*
- *Heart sounds, heartbeat*
- *Otoacoustic emission*
- *Human group actions*

○ Animal sounds

- *Domestic animals, pets*
- *Livestock, farm animals, working animals*
- *Wild animals*

○ Natural sounds

- *Wind*
- *Thunderstorm*

○ Sounds of things

- *Vehicle*
- *Engine*
- *Domestic sounds, home sounds*
- *Bell*
- *Alarm*
- *Mechanisms*
- *Tools*
- *Explosion*
- *Wood*
- *Glass*
- *Liquid*
- *Miscellaneous sources*
- *Specific impact sounds*

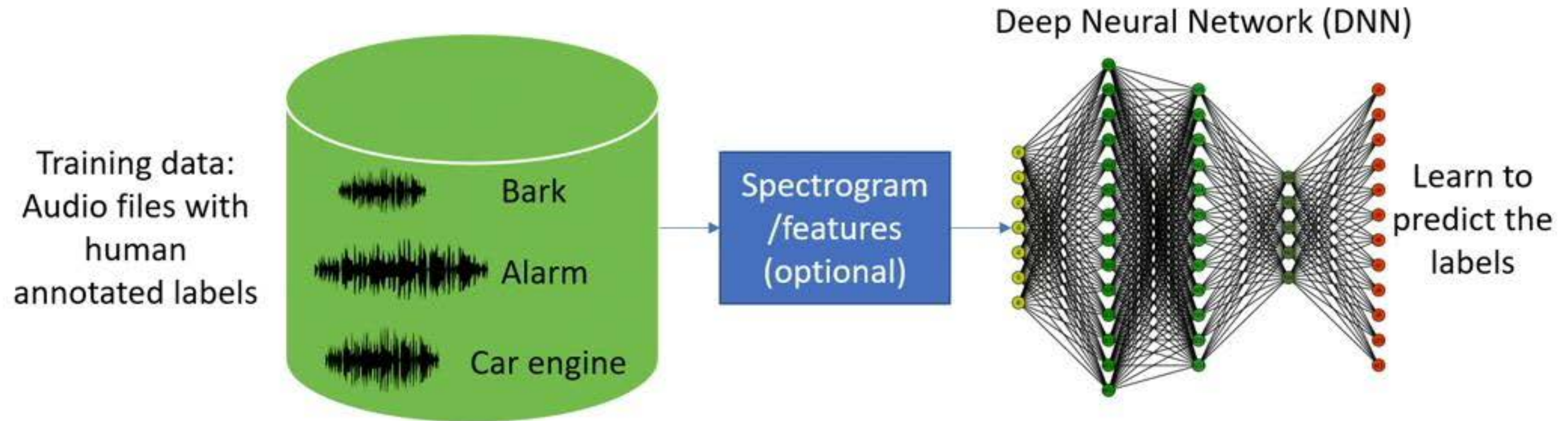
○ Source-ambiguous sounds

- *Generic impact sounds*
- *Surface contact*
- *Deformable shell*

Audio Event Detection & Classification

General machine learning pipeline

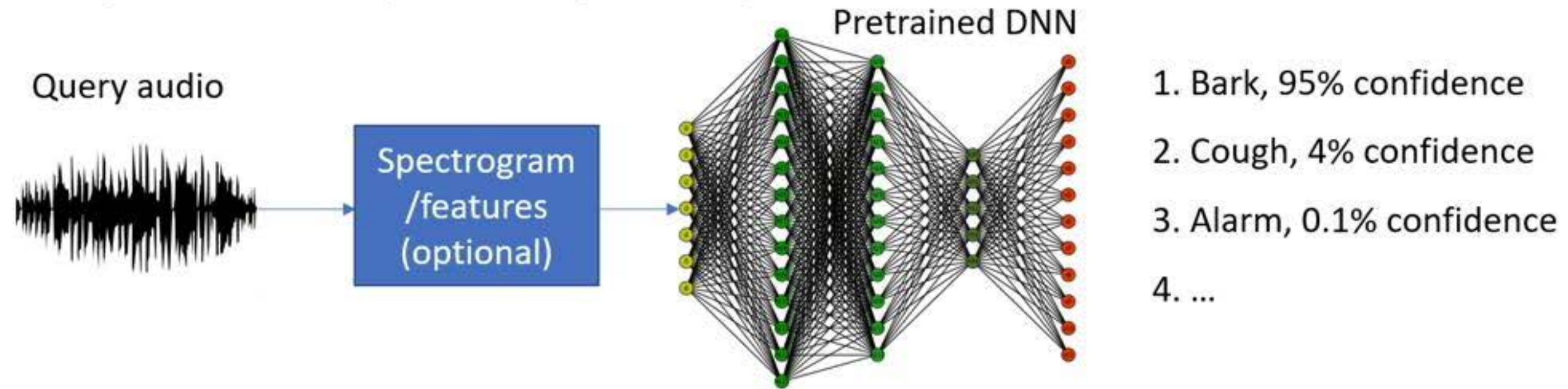
- Training/learning



Audio Event Detection & Classification

General machine learning pipeline

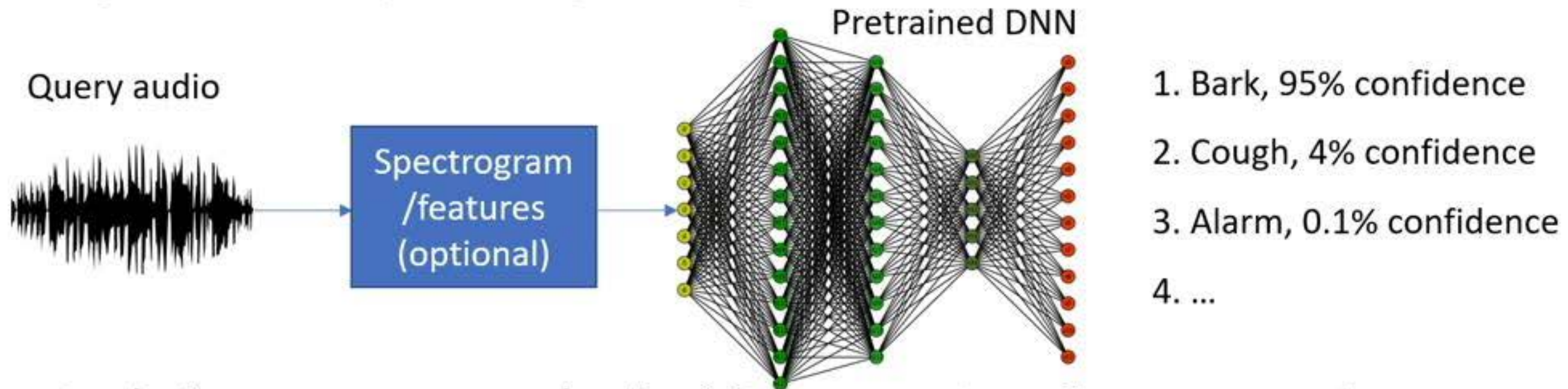
- Testing/inference – *predicting label of a new sound*



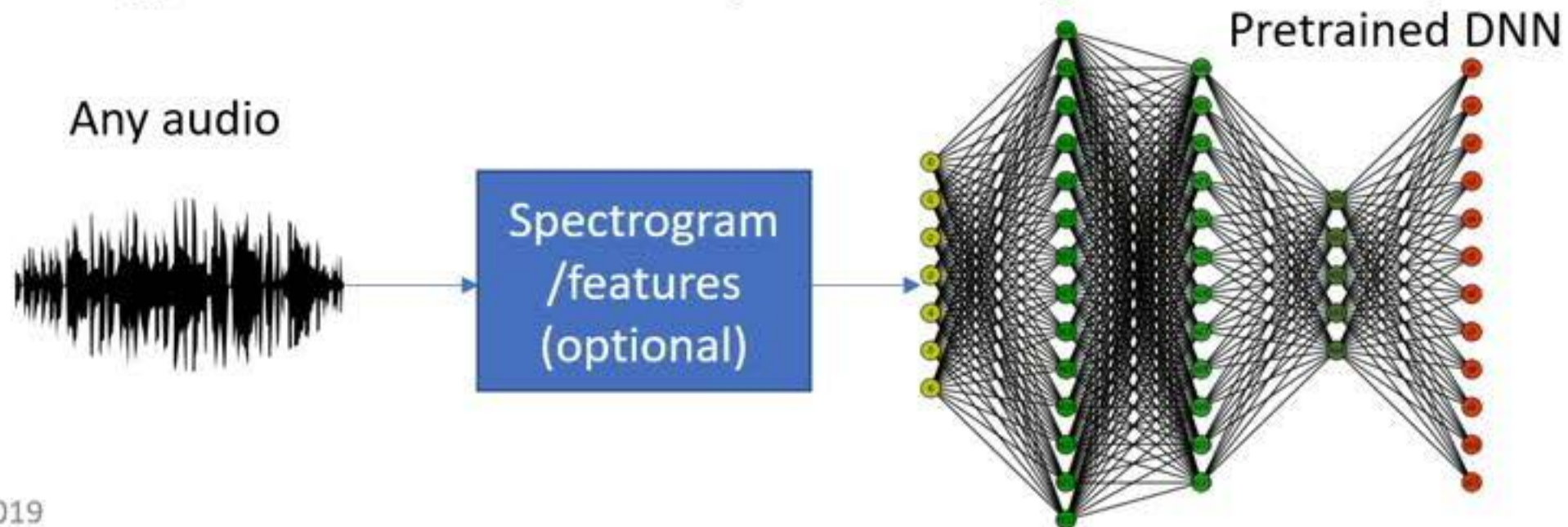
Audio Event Detection & Classification

General machine learning pipeline

- Testing/inference – *predicting label of a new sound*



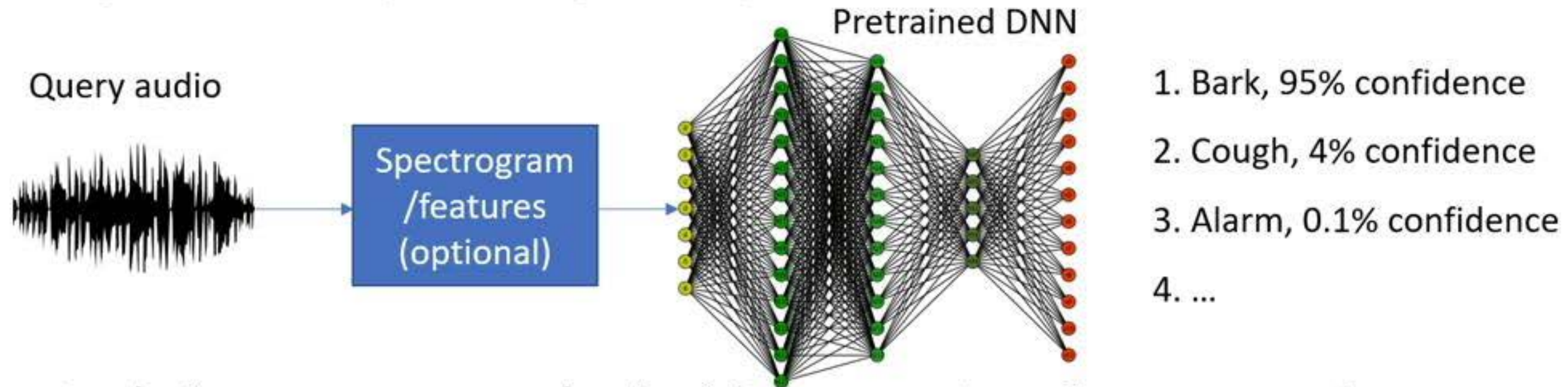
- Testing/inference – *Feature/embedding extraction of a new sound*



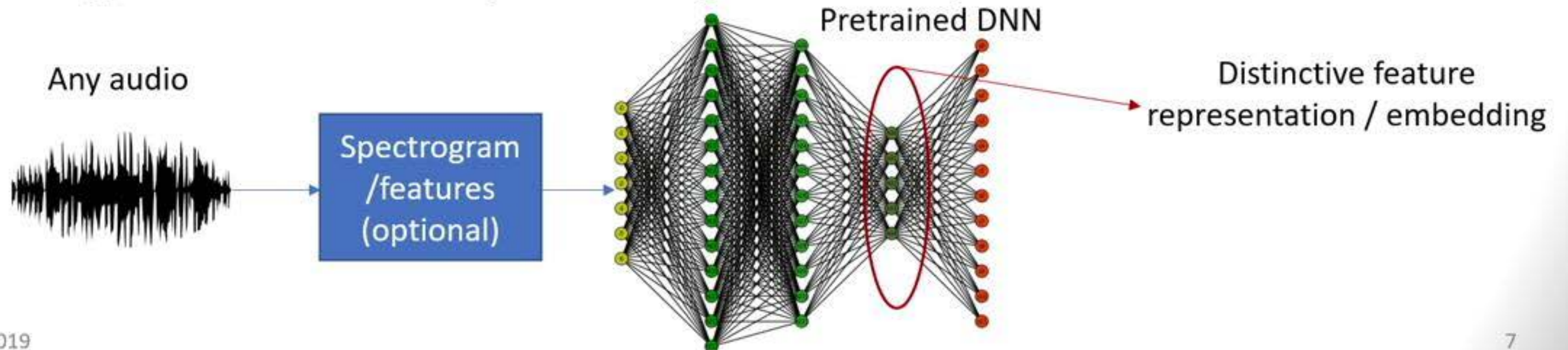
Audio Event Detection & Classification

General machine learning pipeline

- Testing/inference – *predicting label of a new sound*



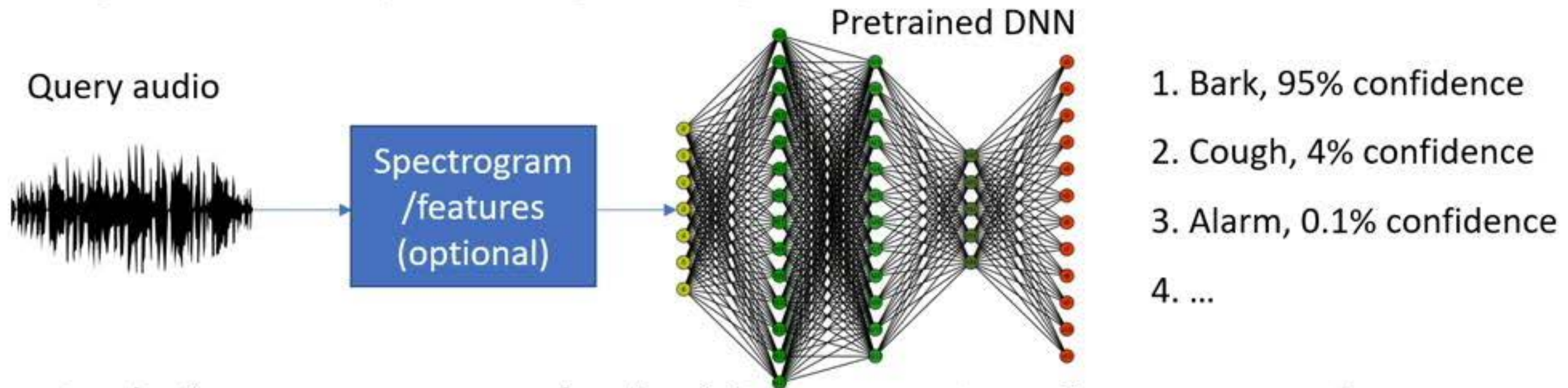
- Testing/inference – *Feature/embedding extraction of a new sound*



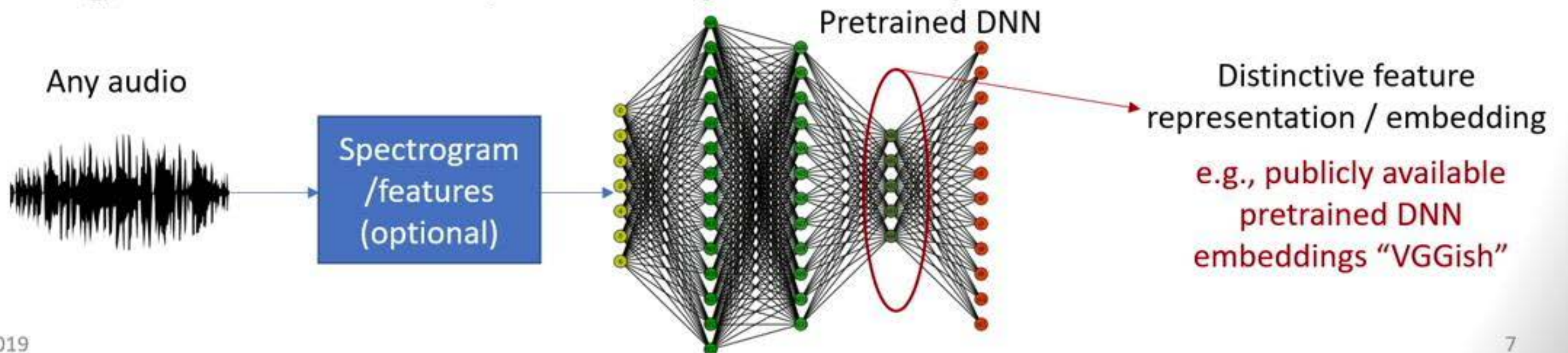
Audio Event Detection & Classification

General machine learning pipeline

- Testing/inference – *predicting label of a new sound*



- Testing/inference – *Feature/embedding extraction of a new sound*



Audio Event Detection & Classification

Applications

- Accessibility
 - Microsoft Soundscape, Hearing AI
- Autonomous driving
- Smart cities, crime prevention, smart home
- Audio content understanding and retrieval
 - Browsing
 - Multimedia synthesis



Audio Event Detection & Classification

Applications

- Accessibility
 - Microsoft Soundscape, Hearing AI
- **Autonomous driving**
- Smart cities, crime prevention, smart home
- Audio content understanding and retrieval
 - Browsing
 - Multimedia synthesis



Audio Event Detection & Classification

Applications

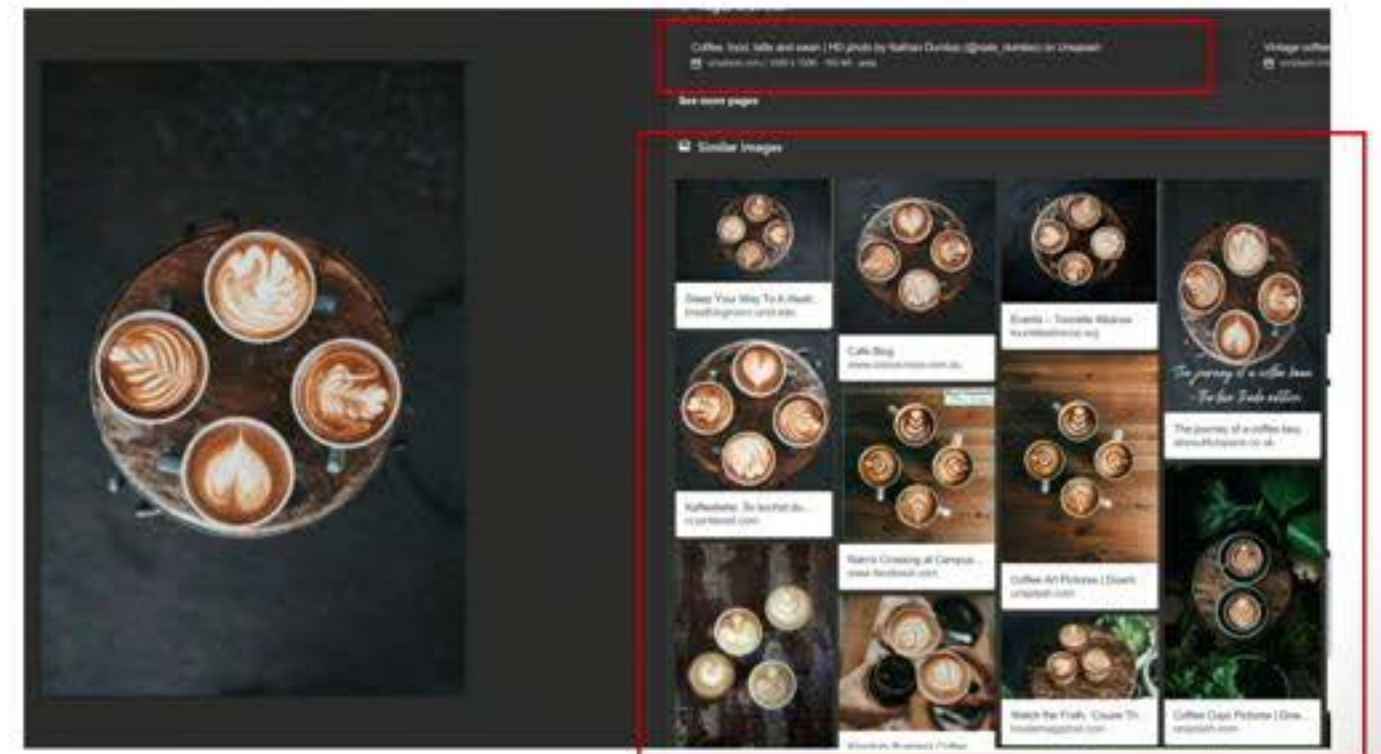
- Accessibility
 - Microsoft Soundscape, Hearing AI
- Autonomous driving
- Smart cities, crime prevention, smart home
- Audio content understanding and retrieval
 - Browsing
 - Multimedia synthesis



Audio Event Detection & Classification

Applications

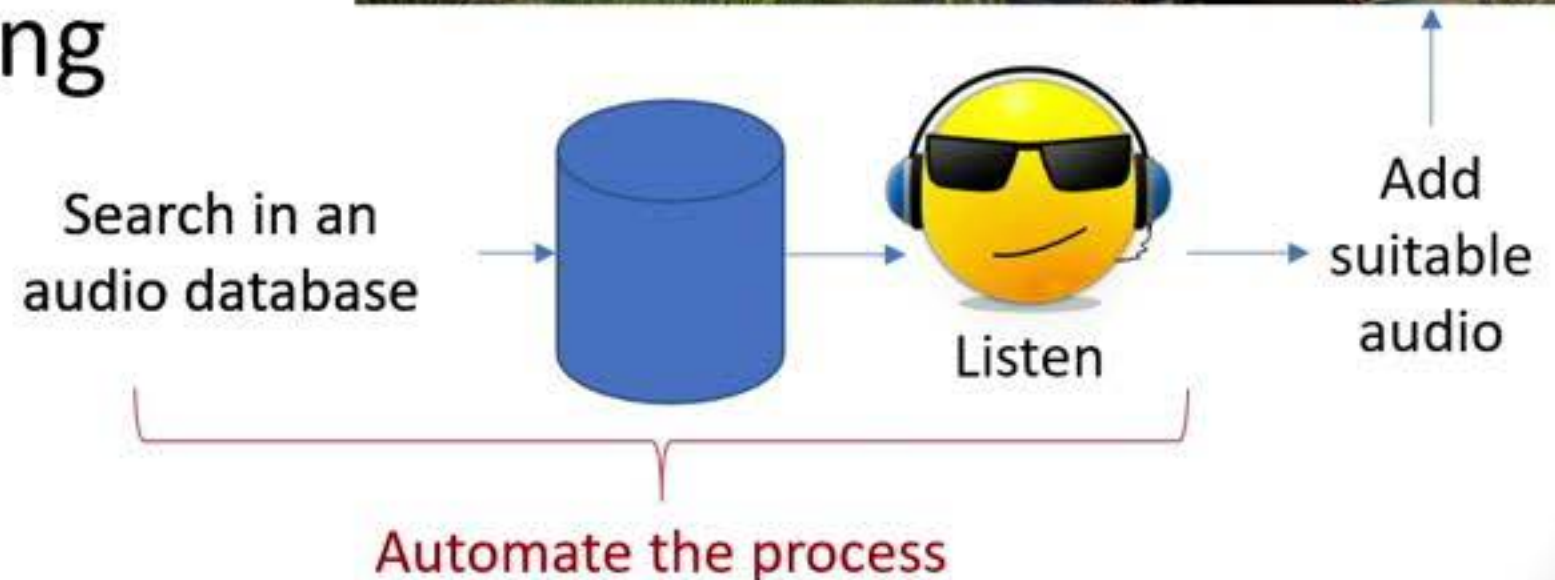
- Accessibility
 - Microsoft Soundscape, Hearing AI
- Autonomous driving
- Smart cities, crime prevention, smart home
- **Audio content understanding and retrieval**
 - Browsing
 - Multimedia synthesis



Audio Event Detection & Classification

Applications

- Accessibility
 - Microsoft Soundscape, Hearing AI
- Autonomous driving
- Smart cities, crime prevention, smart home
- **Audio content understanding and retrieval**
 - Browsing
 - Multimedia synthesis



Audio Event Detection & Classification

Research Areas

Research Area	Better Feature Embeddings	Hierarchical Audio Events	Efficient retrieval and ranking
Task	Learning powerful DNN audio embeddings	<ul style="list-style-type: none">• Explore AE label hierarchy• Back-off to coarse class	<ul style="list-style-type: none">• Fast retrieval / similarity search
Past work	<ul style="list-style-type: none">• Well-explored<ul style="list-style-type: none">• Google's VGGish• Facebook's TLWeak	<ul style="list-style-type: none">• Past work on bi-level hierarchy• Not explored for arbitrary hierarchy	<div style="border: 1px solid red; padding: 5px;">Not well explored for audio events</div>

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Audio Event Detection & Classification

Research Areas

Research Area	Better Feature Embeddings	Hierarchical Audio Events	Efficient retrieval and indexing
Task	Learning powerful DNN audio embeddings	<ul style="list-style-type: none">• Explore AE label hierarchy• Back-off to coarse class	<ul style="list-style-type: none">• Fast retrieval / similarity search
Past work	<ul style="list-style-type: none">• Well-explored<ul style="list-style-type: none">• Google's VGGish• Facebook's TLWeak	<ul style="list-style-type: none">• Past work on bi-level hierarchy• Not explored for arbitrary hierarchy	Not well explored for audio events

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Audio Event Detection & Classification

Research Areas

Research Area	Better Feature Embeddings	Hierarchical Audio Events	Efficient retrieval and ranking
Task	Learning powerful DNN audio embeddings	<ul style="list-style-type: none">• Explore AE label hierarchy• Back-off to coarse class	<ul style="list-style-type: none">• Fast retrieval / similarity search
Past work	<ul style="list-style-type: none">• Well-explored<ul style="list-style-type: none">• Google's VGGish• Facebook's TLWeak	<ul style="list-style-type: none">• Past work on bi-level hierarchy• Not explored for arbitrary hierarchy	Not well explored for audio events

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Audio Event Detection & Classification

Research Areas

Research Area	Better Feature Embeddings	Hierarchical Audio Events	Efficient retrieval and ranking
Task	Learning powerful DNN audio embeddings	<ul style="list-style-type: none">• Explore AE <i>label hierarchy</i>• Back-off to coarse class	<ul style="list-style-type: none">• Fast retrieval / similarity search
Past work	<ul style="list-style-type: none">• Well-explored<ul style="list-style-type: none">• Google's VGGish• Facebook's TLWeak	<ul style="list-style-type: none">• Past work on bi-level hierarchy• Not explored for arbitrary hierarchy	Not well explored for audio events

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Audio Event Detection & Classification

Research Areas

Research Area	Better Feature Embeddings	Hierarchical Audio Events	Efficient retrieval and ranking
Task	Learning powerful DNN audio embeddings	<ul style="list-style-type: none">• Explore AE <i>label hierarchy</i>• Back-off to coarse class	<ul style="list-style-type: none">• Fast retrieval / similarity search
Past work	<ul style="list-style-type: none">• Well-explored<ul style="list-style-type: none">• Google's VGGish• Facebook's TLWeak	<ul style="list-style-type: none">• Past work on bi-level hierarchy• Not explored for arbitrary hierarchy	Not well explored for audio events

- Challenges in efficient retrieval and ranking:
 - Millions of sounds in the database
 - No established meaning of “distance” between sound types or sound events
 - High dimensional feature representations of sounds or DNN embeddings
 - Computation of distance can be highly expensive

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

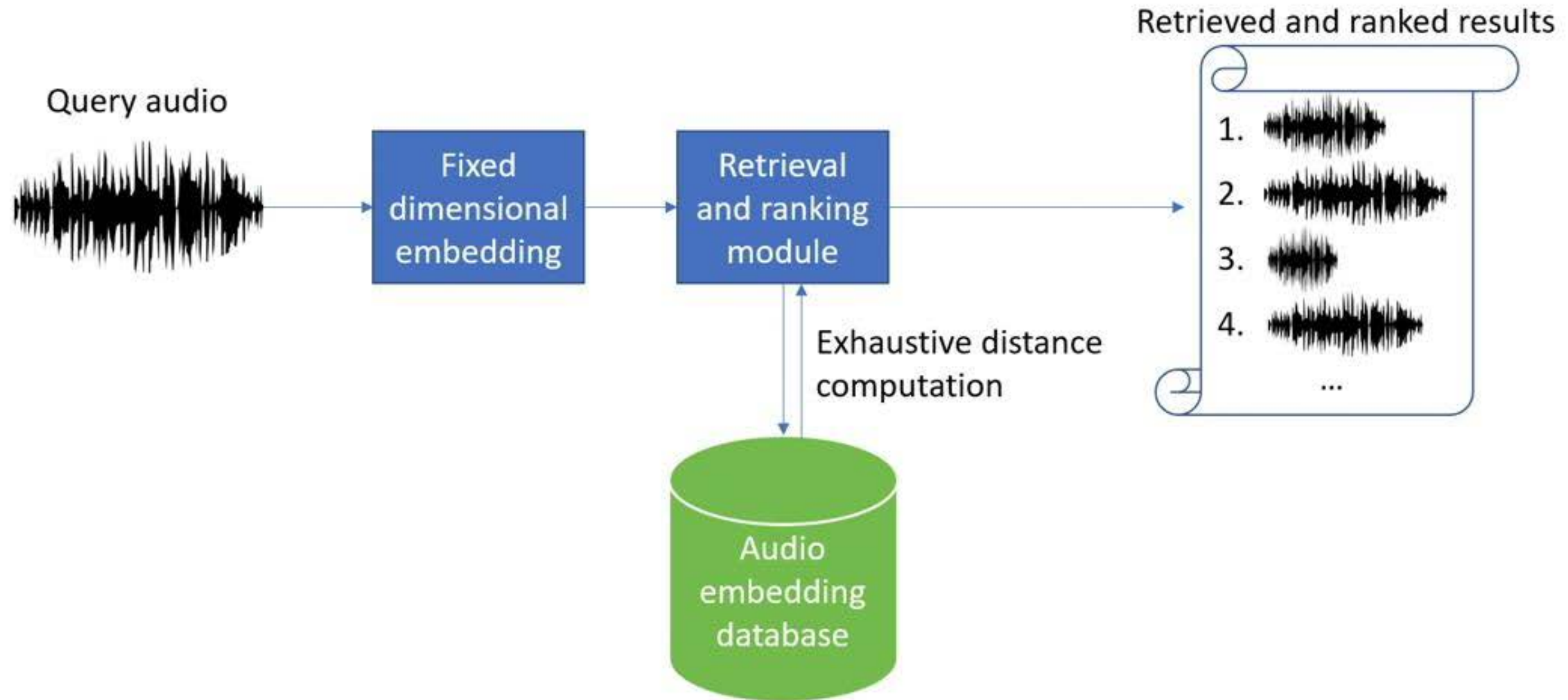
TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Agenda

- Audio event detection & classification
- **Audio retrieval and ranking**
 - Literature review
- Efficient audio retrieval with hashing
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- Conclusions and future work

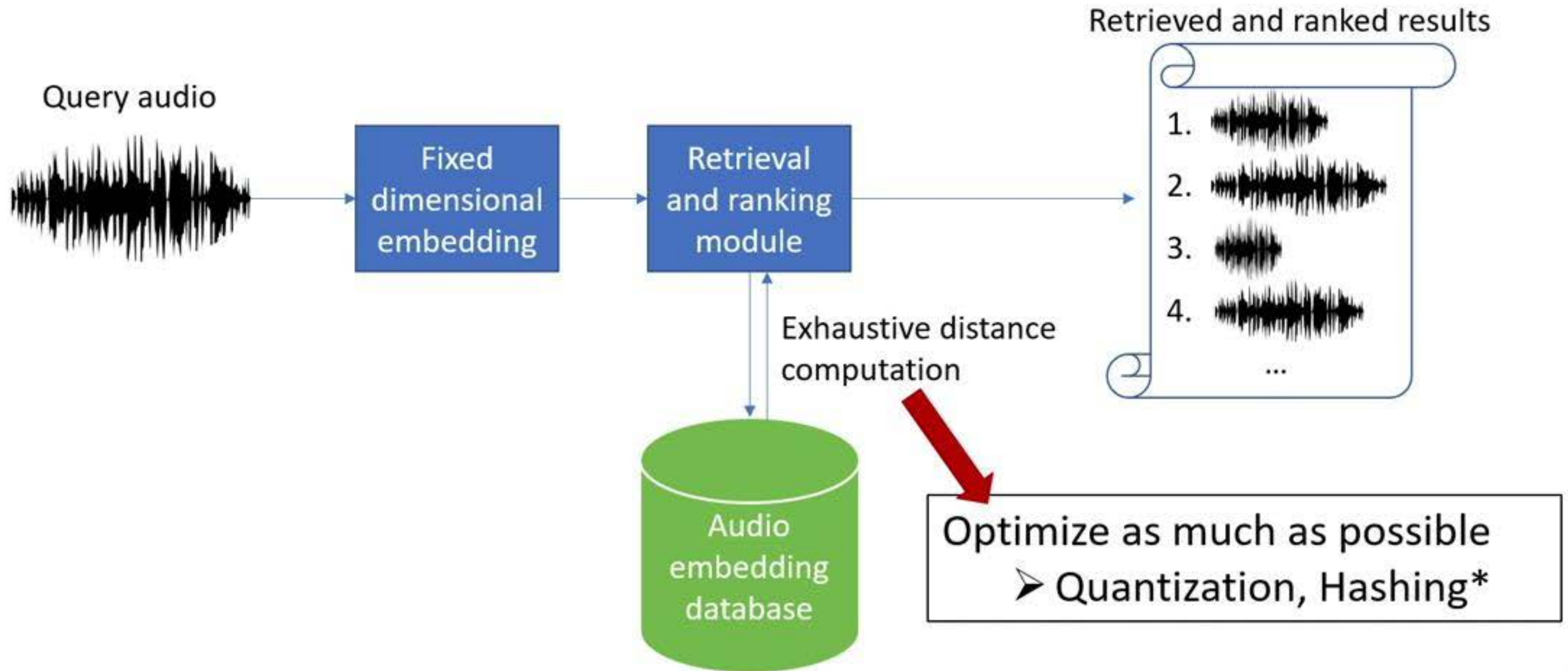
Audio Retrieval and Ranking

High-Level View



Audio Retrieval and Ranking

High-Level View



Audio Retrieval and Ranking

Problem Formulation

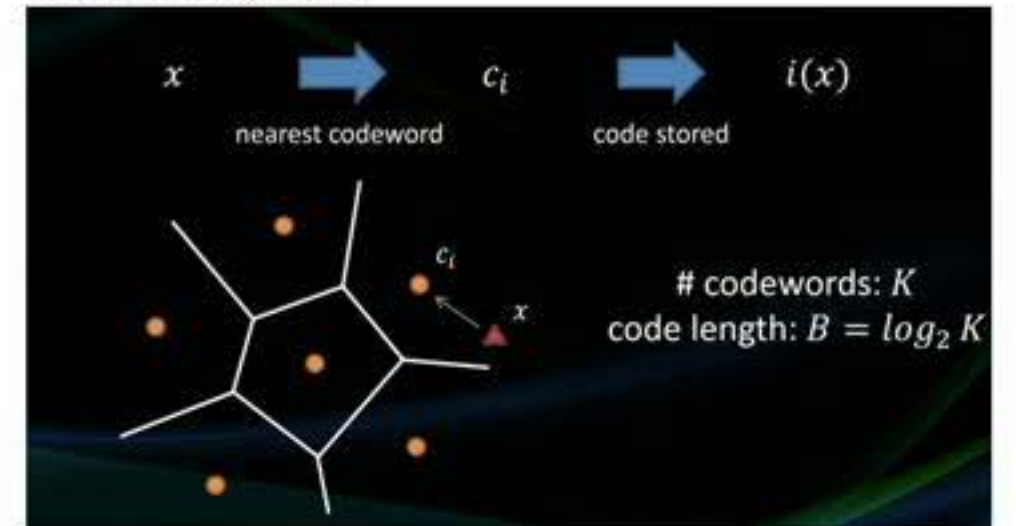
- **Goal:** Efficient distance computation
- **Method:** Approximate nearest neighbors (ANN) search
 - Simple example of Quantization

Audio Retrieval and Ranking

Problem Formulation

- **Goal:** Efficient distance computation
- **Method:** Approximate nearest neighbors (ANN) search
 - Simple example of Quantization

Quantization

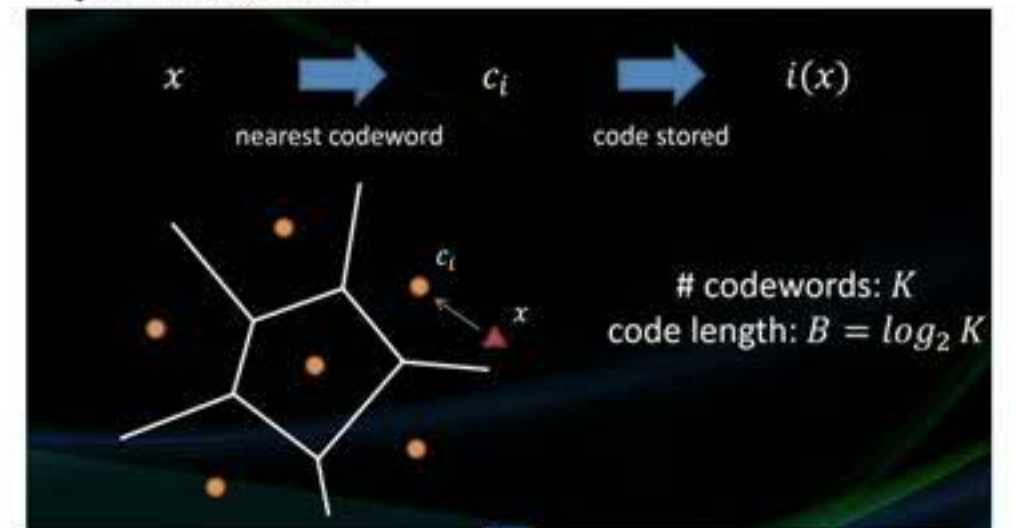


Audio Retrieval and Ranking

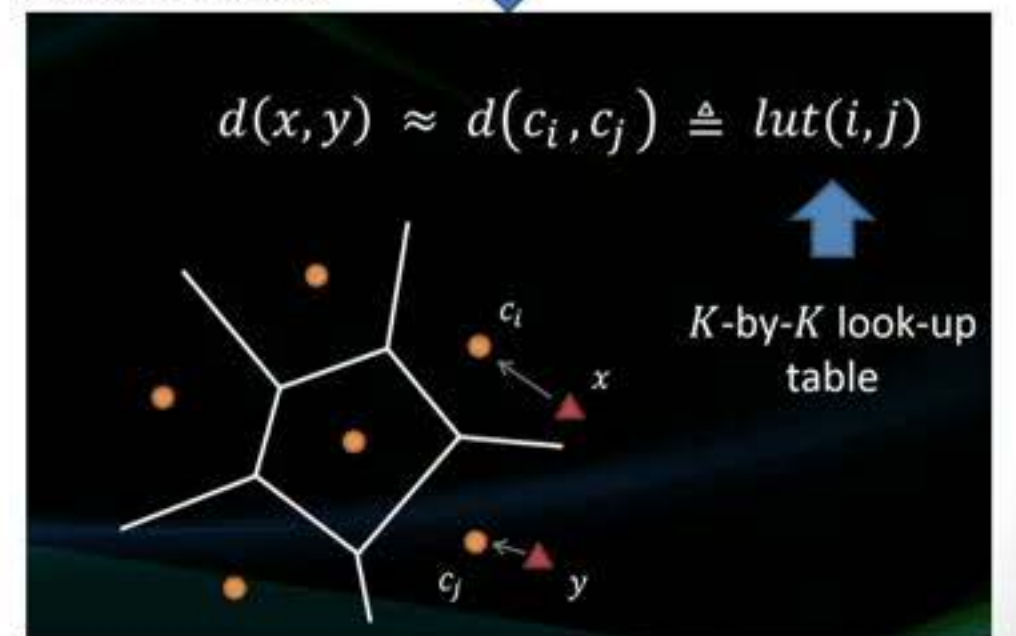
Problem Formulation

- **Goal:** Efficient distance computation
- **Method:** Approximate nearest neighbors (ANN) search
 - Simple example of Quantization

Quantization



ANN search

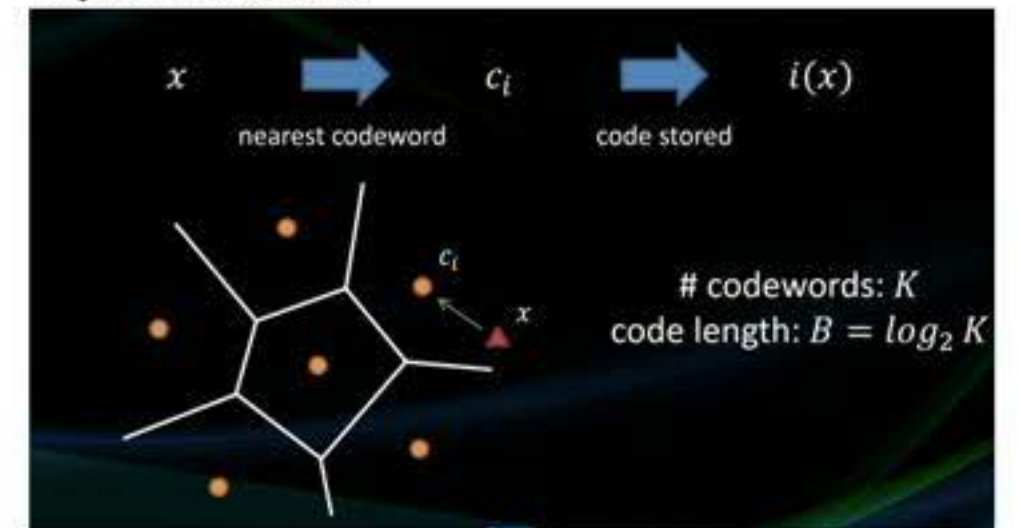


Audio Retrieval and Ranking

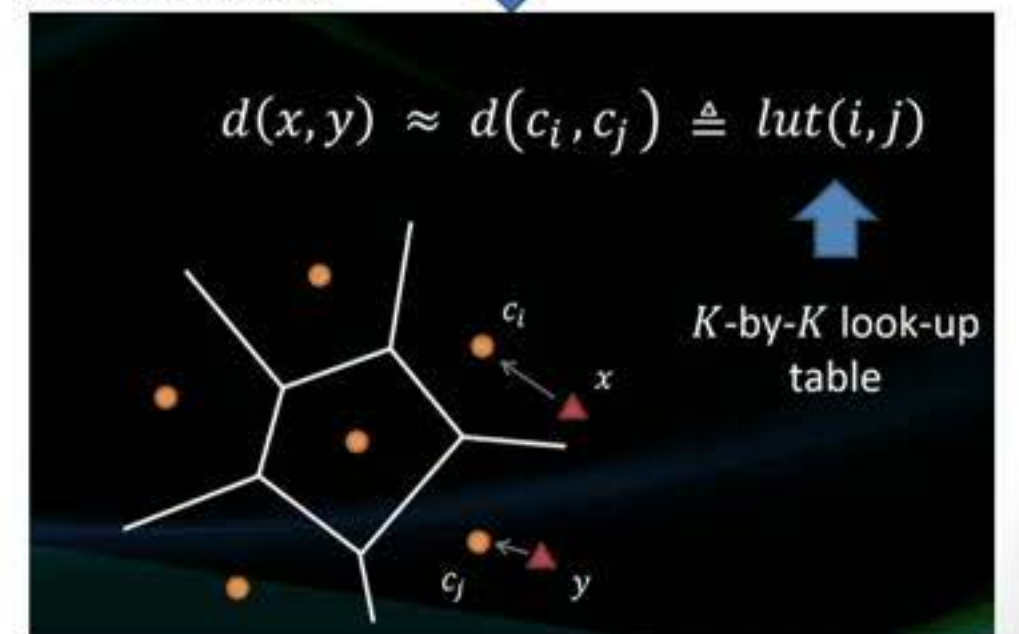
Problem Formulation

- **Goal:** Efficient distance computation
- **Method:** Approximate nearest neighbors (ANN) search
 - Simple example of Quantization
- **Algorithms:**
 - Unsupervised:
 - ☺ • No labels of audio required
 - ☹ • Cannot incorporate human knowledge/semantic meaning

Quantization



ANN search

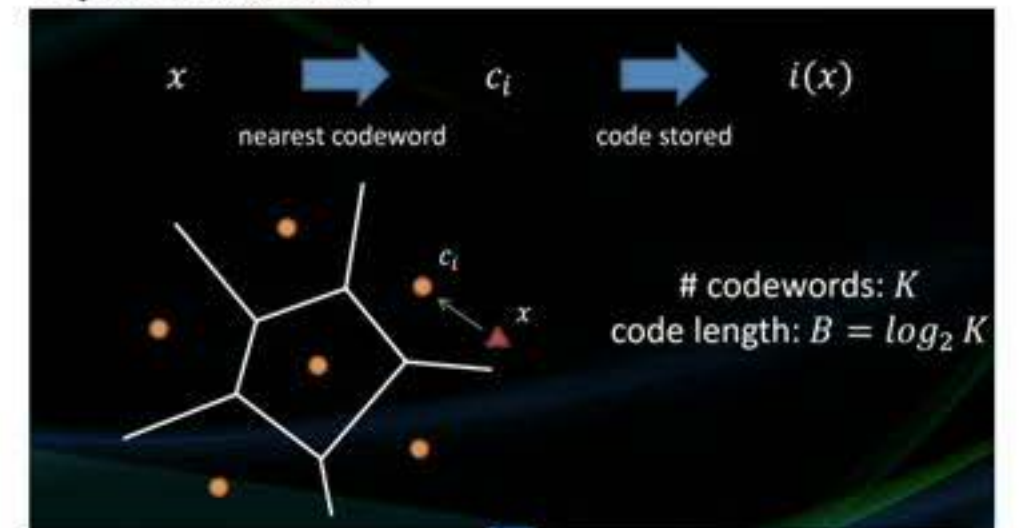


Audio Retrieval and Ranking

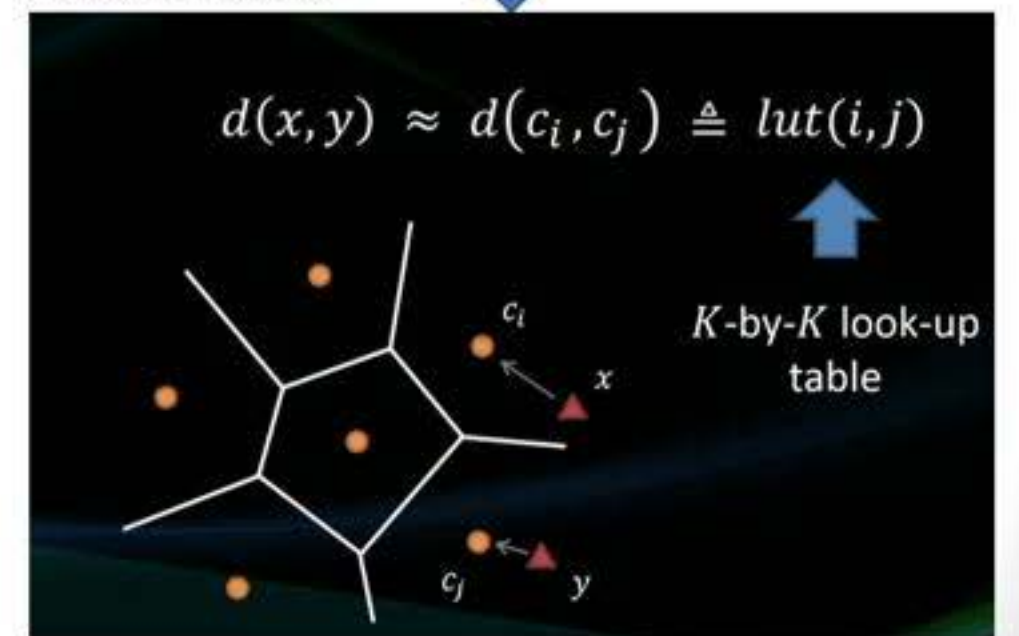
Problem Formulation

- **Goal:** Efficient distance computation
- **Method:** Approximate nearest neighbors (ANN) search
 - Simple example of Quantization
- **Algorithms:**
 - Unsupervised:
 - 😊 • No labels of audio required
 - ☹️ • Cannot incorporate human knowledge/semantic meaning
 - Supervised:
 - 😊 • Exploits human knowledge, preserves data pattern in hash codes
 - ☹️ • Labels of audio required (at least a few examples from the database)

Quantization



ANN search



Agenda

- Audio event detection & classification
- **Audio retrieval and ranking**
 - Literature review
- Efficient audio retrieval with hashing
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- Conclusions and future work

Efficient Retrieval and Ranking

Literature review

Acronym	Title / conference	Authors / Organization	Summary
PQ (Unsupervised)	Product quantization for nearest neighbor search / IEEE PAMI	Jegou, Herve, ... Cordelia Schmid / INRIA Rennes	Unsupervised quantization algorithm, inspired by Vector Quantization and divide & conquer
CNNH (Supervised)	Supervised Hashing for Image Retrieval via Image Representation Learning / AAAI 2014	Rongkai Xia, ..., Shuicheng Yan / NUS Singapore	Stage 1: Learns binary hash codes. Stage 2: Trains a DNN to fit the codes and also class labels.
DNNH (Supervised)	Simultaneous Feature Learning and Hash Coding with Deep Neural Networks / CVPR 2015	Hanjiang Lai, ..., Shuicheng Yan / NUS Singapore	Simultaneous feature learning and hash coding optimized by the triplet loss.
DQN, DTQ (Supervised)	Deep Quantization Network for Efficient Image Retrieval / AAAI 2016 Deep Triplet Quantization, ACM Multimedia 2018	Yue Cao, ..., Jingdong Wang Tsinghua University, China, and Microsoft Research Asia	Joint similarity learning and quantization. Formal control over quantization error. DTQ replaces the pairwise loss with triplet loss.

Efficient Retrieval and Ranking

Literature review

Acronym	Title / conference	Authors / Organization	Summary
PQ (Unsupervised)	Product quantization for nearest neighbor search / IEEE PAMI	Jegou, Herve, ... Cordelia Schmid / INRIA Rennes	Unsupervised quantization algorithm, inspired by Vector Quantization and divide & conquer
CNNH (Supervised)	Supervised Hashing for Image Retrieval via Image Representation Learning / AAAI 2014	Rongkai Xia, ..., Shuicheng Yan / NUS Singapore	Stage 1: Learns binary hash codes. Stage 2: Trains a DNN to fit the codes and also class labels.
DNNH (Supervised)	Simultaneous Feature Learning and Hash Coding with Deep Neural Networks / CVPR 2015	Hanjiang Lai, ..., Shuicheng Yan / NUS Singapore	Simultaneous feature learning and hash coding optimized by the triplet loss.
DQN, DTQ (Supervised)	Deep Quantization Network for Efficient Image Retrieval / AAAI 2016 Deep Triplet Quantization, ACM Multimedia 2018	Yue Cao, ..., Jingdong Wang Tsinghua University, China, and Microsoft Research Asia	Joint similarity learning and quantization. Formal control over quantization error. DTQ replaces the pairwise loss with triplet loss.

Efficient Retrieval and Ranking

Literature review

Acronym	Title / conference	Authors / Organization	Summary
PQ (Unsupervised)	Product quantization for nearest neighbor search / IEEE PAMI	Jegou, Herve, ... Cordelia Schmid / INRIA Rennes	Unsupervised quantization algorithm, inspired by Vector Quantization and divide & conquer
CNNH (Supervised)	Supervised Hashing for Image Retrieval via Image Representation Learning / AAAI 2014	Rongkai Xia, ..., Shuicheng Yan / NUS Singapore	Stage 1: Learns binary hash codes. Stage 2: Trains a DNN to fit the codes and also class labels.
DNNH (Supervised)	Simultaneous Feature Learning and Hash Coding with Deep Neural Networks / CVPR 2015	Hanjiang Lai, ..., Shuicheng Yan / NUS Singapore	Simultaneous feature learning and hash coding optimized by the triplet loss.
DQN, DTQ (Supervised)	Deep Quantization Network for Efficient Image Retrieval / AAAI 2016 Deep Triplet Quantization, ACM Multimedia 2018	Yue Cao, ..., Jingdong Wang Tsinghua University, China, and Microsoft Research Asia	Joint similarity learning and quantization. Formal control over quantization error. DTQ replaces the pairwise loss with triplet loss.

Efficient Retrieval and Ranking

Literature review

Acronym	Title / conference	Authors / Organization	Summary
PQ (Unsupervised)	Product quantization for nearest neighbor search / IEEE PAMI	Jegou, Herve, ... Cordelia Schmid / INRIA Rennes	Unsupervised quantization algorithm, inspired by Vector Quantization and divide & conquer
CNNH (Supervised)	Supervised Hashing for Image Retrieval via Image Representation Learning / AAAI 2014	Rongkai Xia, ..., Shuicheng Yan / NUS Singapore	Stage 1: Learns binary hash codes. Stage 2: Trains a DNN to fit the codes and also class labels.
DNNH (Supervised)	Simultaneous Feature Learning and Hash Coding with Deep Neural Networks / CVPR 2015	Hanjiang Lai, ..., Shuicheng Yan / NUS Singapore	Simultaneous feature learning and hash coding optimized by the triplet loss.
DQN, DTQ (Supervised)	Deep Quantization Network for Efficient Image Retrieval / AAAI 2016 Deep Triplet Quantization, ACM Multimedia 2018	Yue Cao, ..., Jingdong Wang Tsinghua University, China, and Microsoft Research Asia	Joint similarity learning and quantization. Formal control over quantization error. DTQ replaces the pairwise loss with triplet loss.

Efficient Retrieval and Ranking

Literature review

Acronym	Title / conference	Authors / Organization	Summary
PQ (Unsupervised)	Product quantization for nearest neighbor search / IEEE PAMI	Jegou, Herve, ... Cordelia Schmid / INRIA Rennes	Unsupervised quantization algorithm, inspired by Vector Quantization and divide & conquer
CNNH (Supervised)	Supervised Hashing for Image Retrieval via Image Representation Learning / AAAI 2014	Rongkai Xia, ..., Shuicheng Yan / NUS Singapore	Stage 1: Learns binary hash codes. Stage 2: Trains a DNN to fit the codes and also class labels.
DNNH (Supervised)	Simultaneous Feature Learning and Hash Coding with Deep Neural Networks / CVPR 2015	Hanjiang Lai, ..., Shuicheng Yan / NUS Singapore	Simultaneous feature learning and hash coding optimized by the triplet loss.
DQN, DTQ (Supervised)	Deep Quantization Network for Efficient Image Retrieval / AAAI 2016 Deep Triplet Quantization, ACM Multimedia 2018	Yue Cao, ..., Jingdong Wang Tsinghua University, China, and Microsoft Research Asia	Joint similarity learning and quantization. Formal control over quantization error. DTQ replaces the pairwise loss with triplet loss.

Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- **Efficient audio retrieval with hashing**
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- Conclusions and future work

Efficient Audio Retrieval with Hashing

Preliminary

- Nearest Neighbor search with Euclidean distance

- $N = \#$ samples in the database

- $D =$ feature dimension

- Nearest neighbor of a query x :

$\operatorname{argmin} \operatorname{dist}(x, y_i)$

$\forall i = 1, \dots, N$

- Complexity:

$\mathcal{O}(ND)$

Efficient Audio Retrieval with Hashing

Preliminary

- Nearest Neighbor search with Euclidean distance

- $N = \#$ samples in the database

- $D =$ feature dimension

- Nearest neighbor of a query x :

$$\operatorname{argmin}_{i} \operatorname{dist}(x, y_i)$$

$$\forall i = 1, \dots, N$$

- Complexity:

$$\mathcal{O}(ND)$$

- Example

- $N = \#$ samples in the database

$$= 1M$$

- $D =$ feature dimension

$$= 1000$$

- Complexity:

$$= \mathcal{O}(1B)$$

Efficient Audio Retrieval with Hashing

Preliminary

- Nearest Neighbor search with Euclidean distance

- $N = \#$ samples in the database

- $D =$ feature dimension

- Nearest neighbor of a query x :

$$\operatorname{argmin}_{i} \operatorname{dist}(x, y_i)$$

$$\forall i = 1, \dots, N$$

- Complexity:

$$\mathcal{O}(ND)$$

- Example

- $N = \#$ samples in the database

$$= 1M$$

- $D =$ feature dimension

$$= 1000$$

- Complexity:

$$= \mathcal{O}(1B)$$

Also known as “Curse of dimensionality”

Efficient Audio Retrieval with Hashing

Preliminary

- Nearest Neighbor search with Euclidean distance

- $N = \#$ samples in the database

- $D =$ feature dimension

- Nearest neighbor of a query x :

$$\operatorname{argmin}_{i} \operatorname{dist}(x, y_i)$$

$$\forall i = 1, \dots, N$$

- Complexity:

$$\mathcal{O}(ND)$$

- Example

- $N = \#$ samples in the database

$$= 1M$$

- $D =$ feature dimension

$$= 1000$$

- Complexity:

$$= \mathcal{O}(1B)$$

Also known as “Curse of dimensionality”

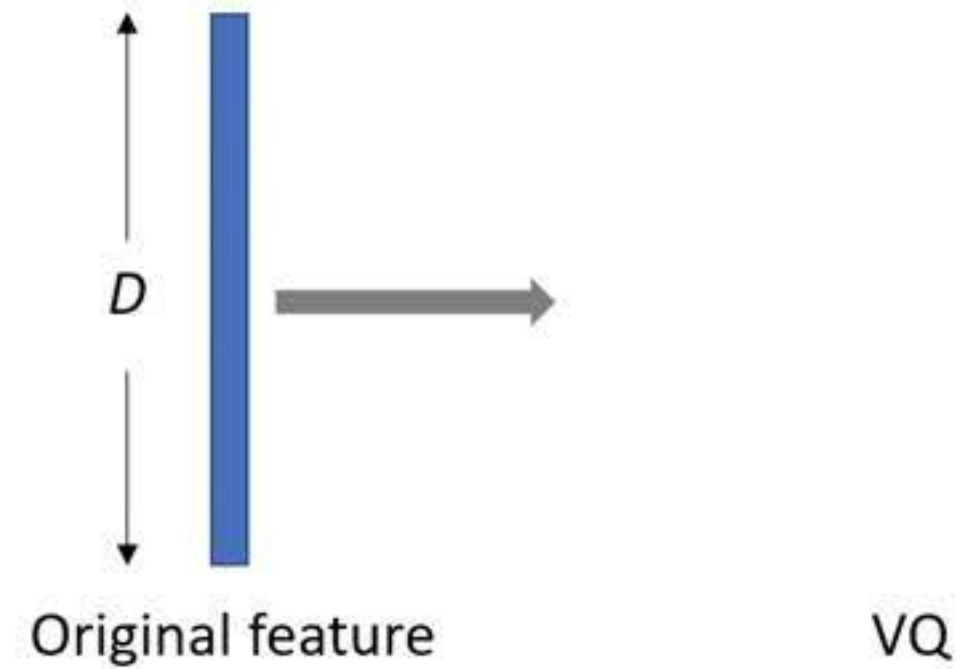
- We will achieve via hashing:

$$\sim \mathcal{O}(9M)$$

Unsupervised Hashing Algorithms

Preliminary – Vector Quantization (VQ)

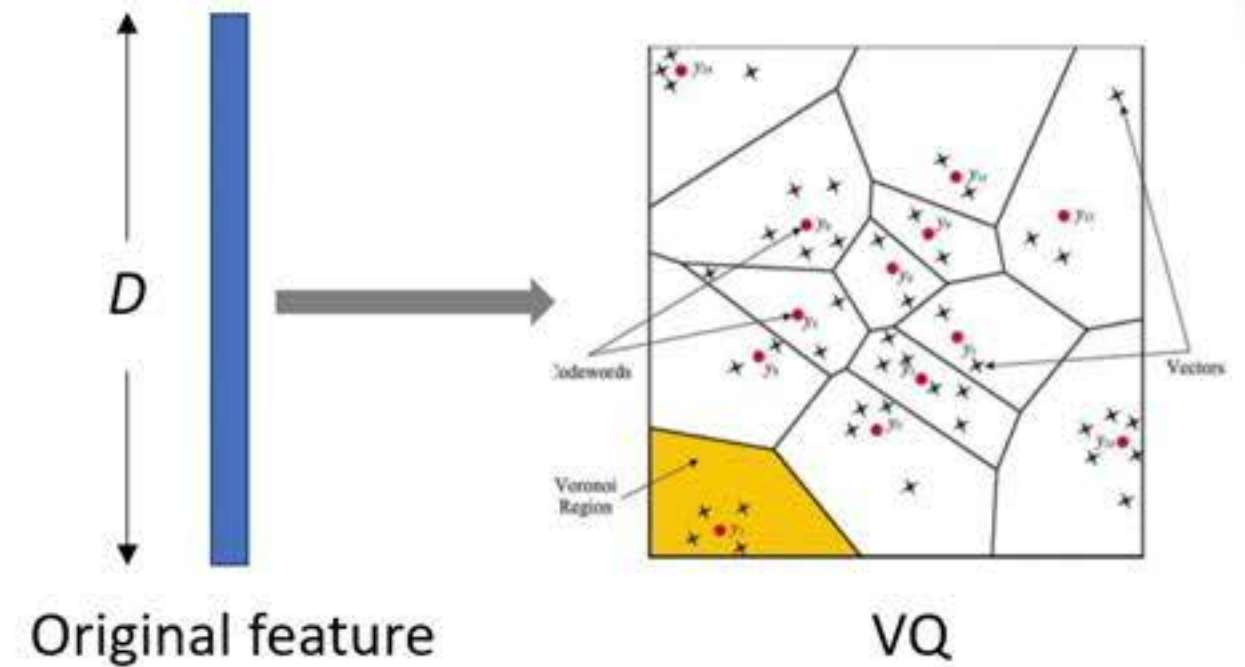
- Goal: Sample \rightarrow Codeword



Unsupervised Hashing Algorithms

Preliminary – Vector Quantization (VQ)

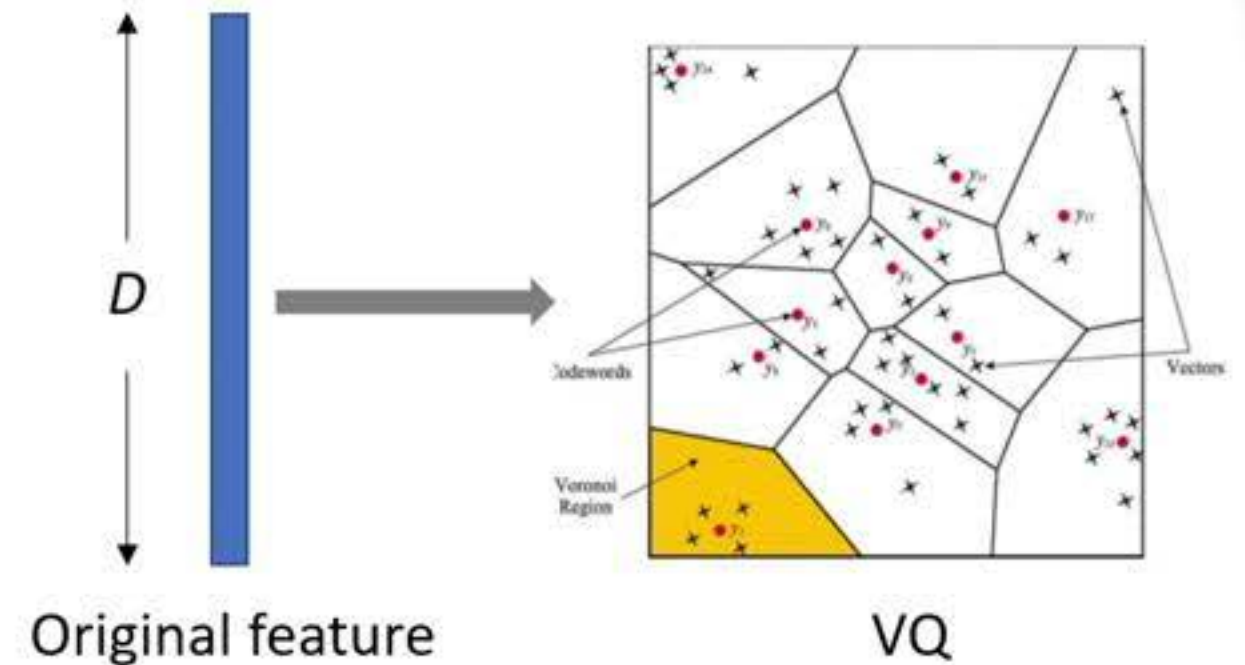
- Goal: Sample \rightarrow Codeword
- Method:
 - Step 1: K-Mean clustering:
Centroids = codewords
 - Step 2: Quantization: Sample \approx Closest centroid



Unsupervised Hashing Algorithms

Preliminary – Vector Quantization (VQ)

- Goal: Sample \rightarrow Codeword
- Method:
 - Step 1: K-Mean clustering:
Centroids = codewords
 - Step 2: Quantization: Sample \approx Closest centroid
- K centroids/codewords $\rightarrow \log_2 K$ bits for each vector
 - e.g., for K=8 centroids, we need 3 bits



Unsupervised Hashing Algorithm

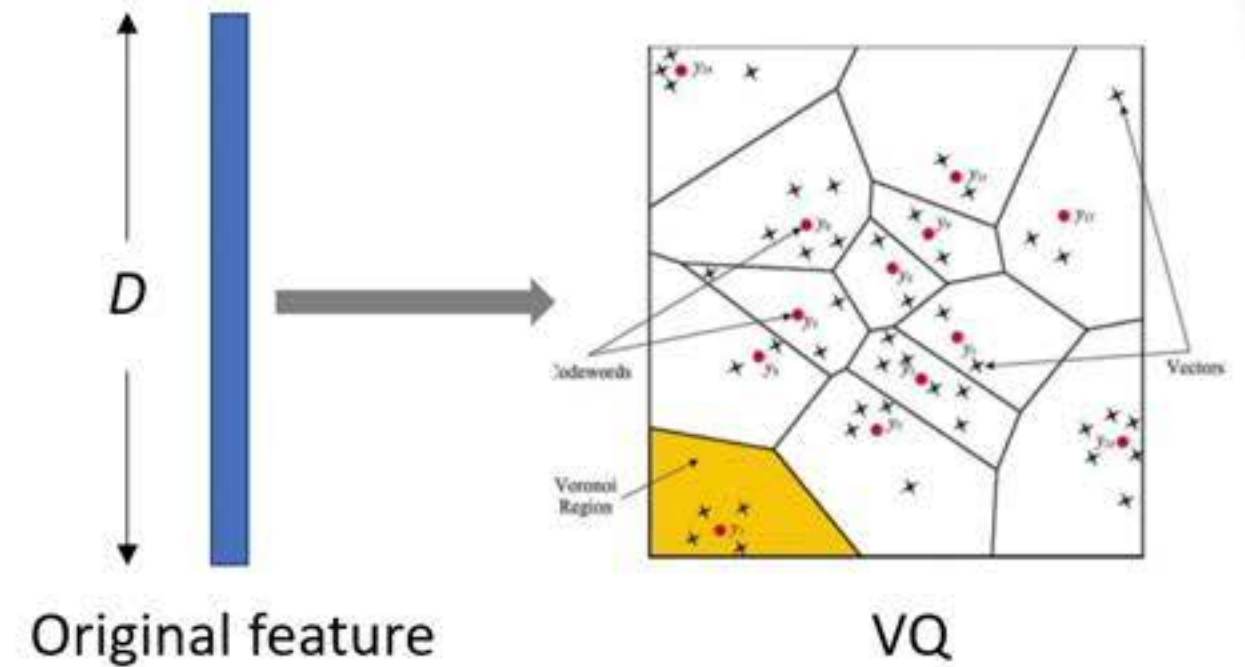
Product Quantization (PQ)

- Motivation:
 - Vector Quantization (VQ) cannot support exponentially large number of codewords

Unsupervised Hashing Algorithms

Preliminary – Vector Quantization (VQ)

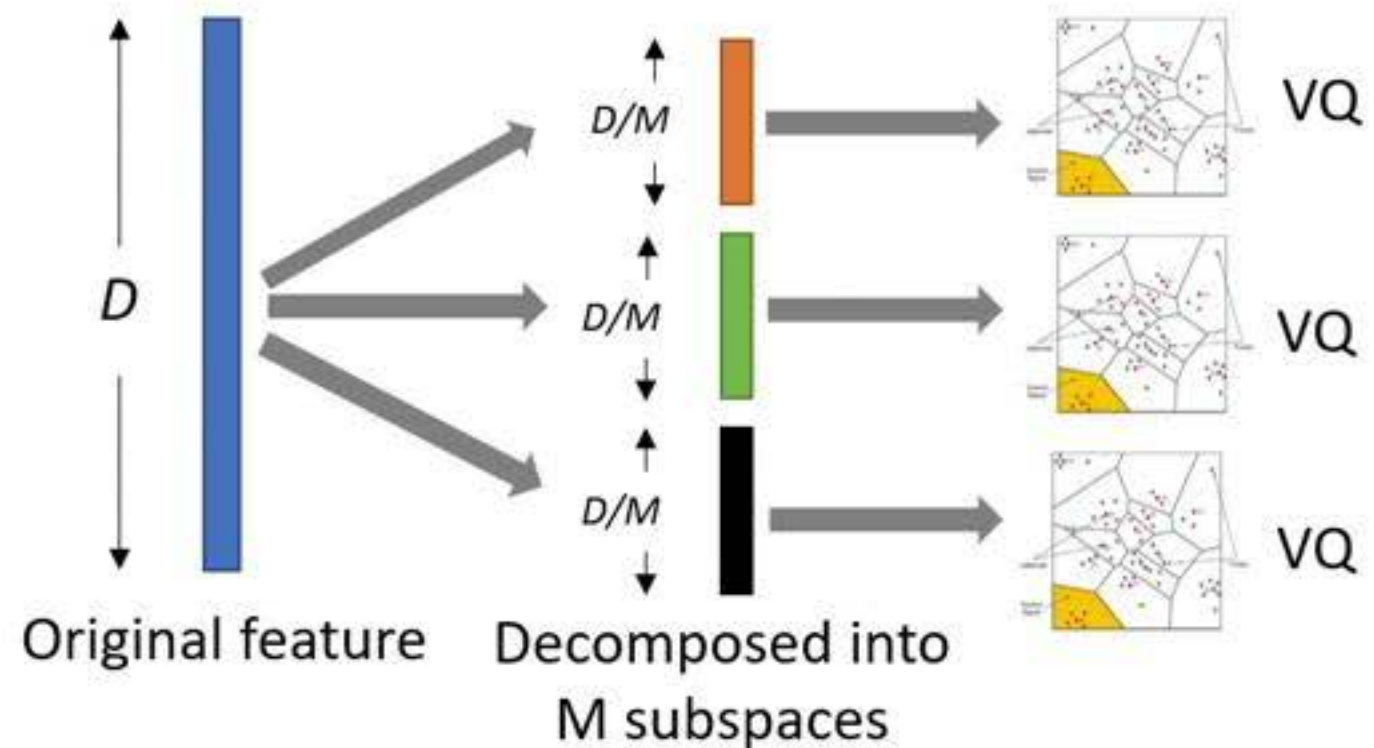
- Goal: Sample \rightarrow Codeword
- Method:
 - Step 1: K-Mean clustering:
Centroids = codewords
 - Step 2: Quantization: Sample \approx Closest centroid
- K centroids/codewords $\rightarrow \log_2 K$ bits for each vector
 - e.g., for K=8 centroids, we need 3 bits



Unsupervised Hashing Algorithm

Product Quantization (PQ)

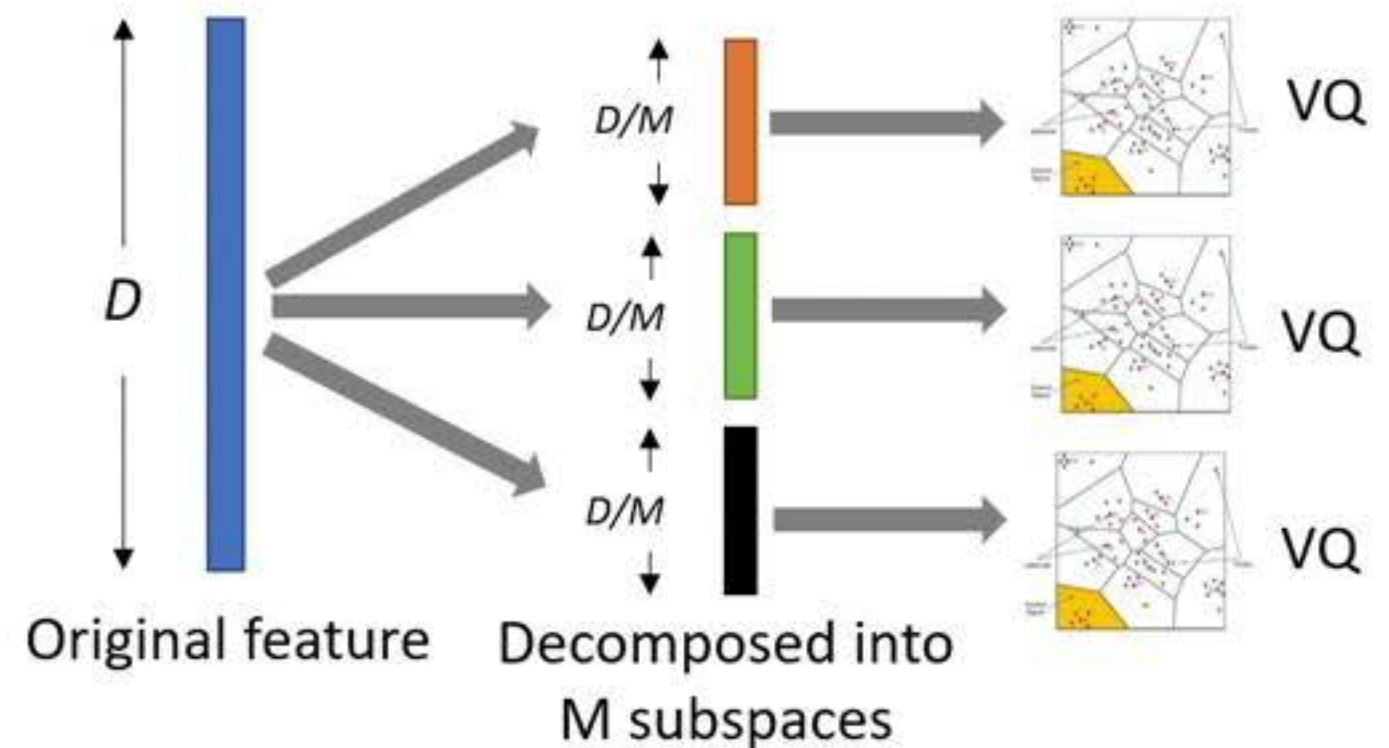
- Motivation:
 - Vector Quantization (VQ) cannot support exponentially large number of codewords
- PQ Algorithm:
 - Decompose the space into a Cartesian product of low-dimensional subspaces
 - Quantize each subspace separately – using VQ



Unsupervised Hashing Algorithm

Product Quantization (PQ)



- Motivation:
 - Vector Quantization (VQ) cannot support exponentially large number of codewords
- PQ Algorithm:
 - Decompose the space into a Cartesian product of low-dimensional subspaces
 - Quantize each subspace separately – using VQ



- **M subspaces**, each of dimension $D_{subspace} = D/M$
- Set, $K_{subspace}$ centroids per subspace
- Typical, $K_{subspace} = 256$
- Effective codebook, $C = \{C_1 \times C_2 \times C_3 \times \dots \times C_M\}$
- Effective codebook size, $K = (K_{subspace})^M$

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

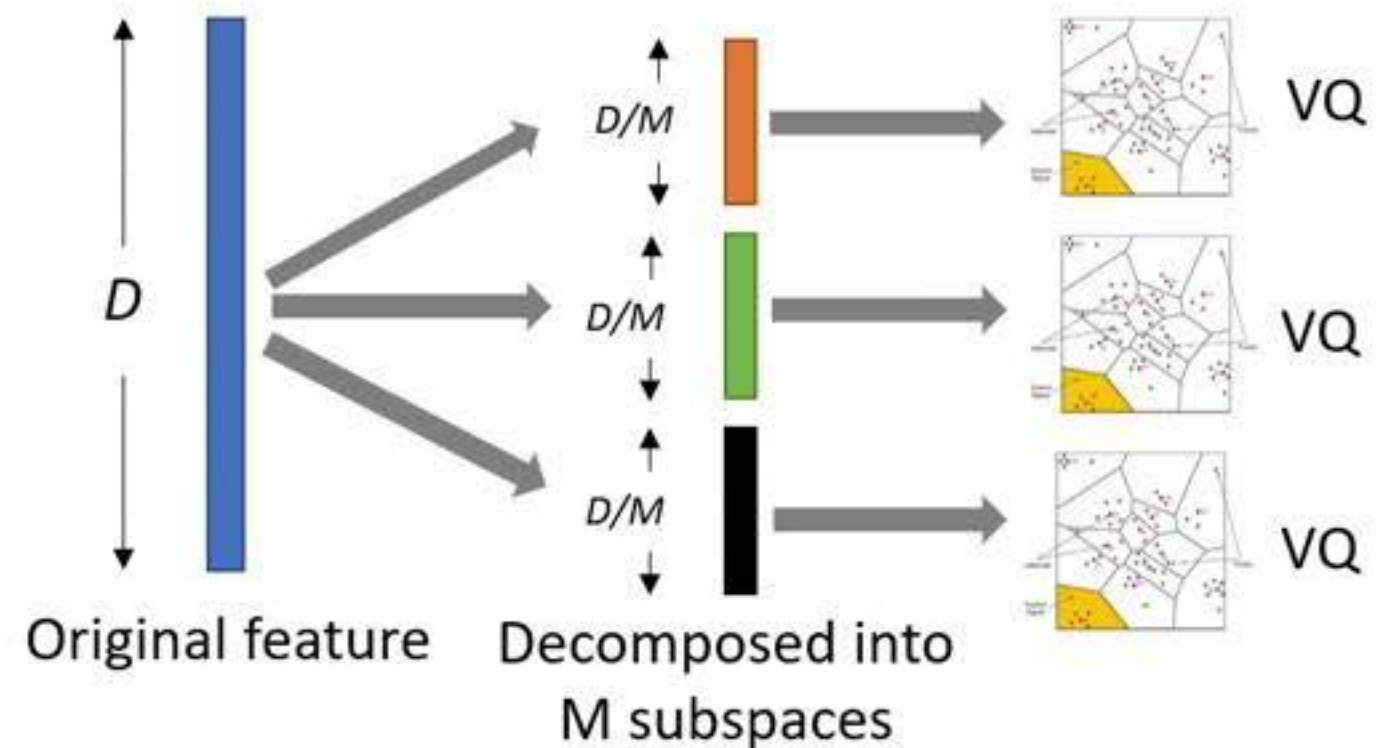
Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$?
Effective # codewords	--	K	$(K_{subspace})^M$?
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

Unsupervised Hashing Algorithm

Product Quantization (PQ)



- Motivation:
 - Vector Quantization (VQ) cannot support exponentially large number of codewords
- PQ Algorithm:
 - Decompose the space into a Cartesian product of low-dimensional subspaces
 - Quantize each subspace separately – using VQ



- **M subspaces**, each of dimension $D_{subspace} = D/M$
- Set, $K_{subspace}$ centroids per subspace
- Typical, $K_{subspace} = 256$
- Effective codebook, $C = \{C_1 \times C_2 \times C_3 \times \dots \times C_M\}$
- Effective codebook size, $K = (K_{subspace})^M$

Euclidean → VQ → PQ → DQN

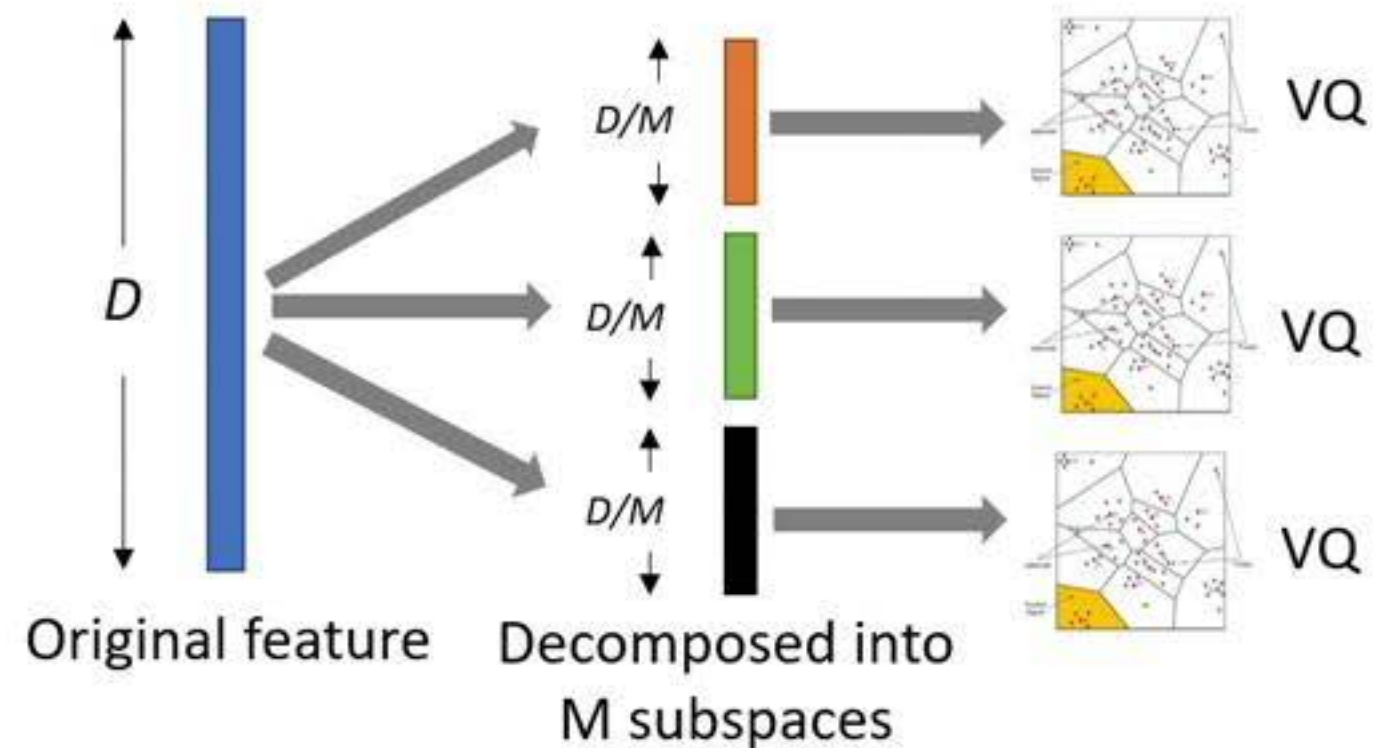
Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$?
Effective # codewords	--	K	$(K_{subspace})^M$?
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K ⇒ high error	Cannot retain data pattern in hash codes	Needs some labeled data

Unsupervised Hashing Algorithm

Product Quantization (PQ)



- Motivation:
 - Vector Quantization (VQ) cannot support exponentially large number of codewords
- PQ Algorithm:
 - Decompose the space into a Cartesian product of low-dimensional subspaces
 - Quantize each subspace separately – using VQ



- **M subspaces**, each of dimension $D_{subspace} = D/M$
- Set, $K_{subspace}$ centroids per subspace
- Typical, $K_{subspace} = 256$
- Effective codebook, $C = \{C_1 \times C_2 \times C_3 \times \dots \times C_M\}$
- Effective codebook size, $K = (K_{subspace})^M$



Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$?
Effective # codewords	--	K	$(K_{subspace})^M$?
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN



Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$?
Effective # codewords	--	K	$(K_{subspace})^M$?
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

N = # samples in the database = 1M
 D = feature dimension = 1000
 M = # of subspaces = 8
 # centroids per subspace, $K_{subspace}$ = 256
 $\Rightarrow M \log_2 K_{subspace}$ = 64 bits hash code
 Effective K in PQ, $K = (K_{subspace})^M$ = 256^8

Euclidean → VQ → PQ → DQN

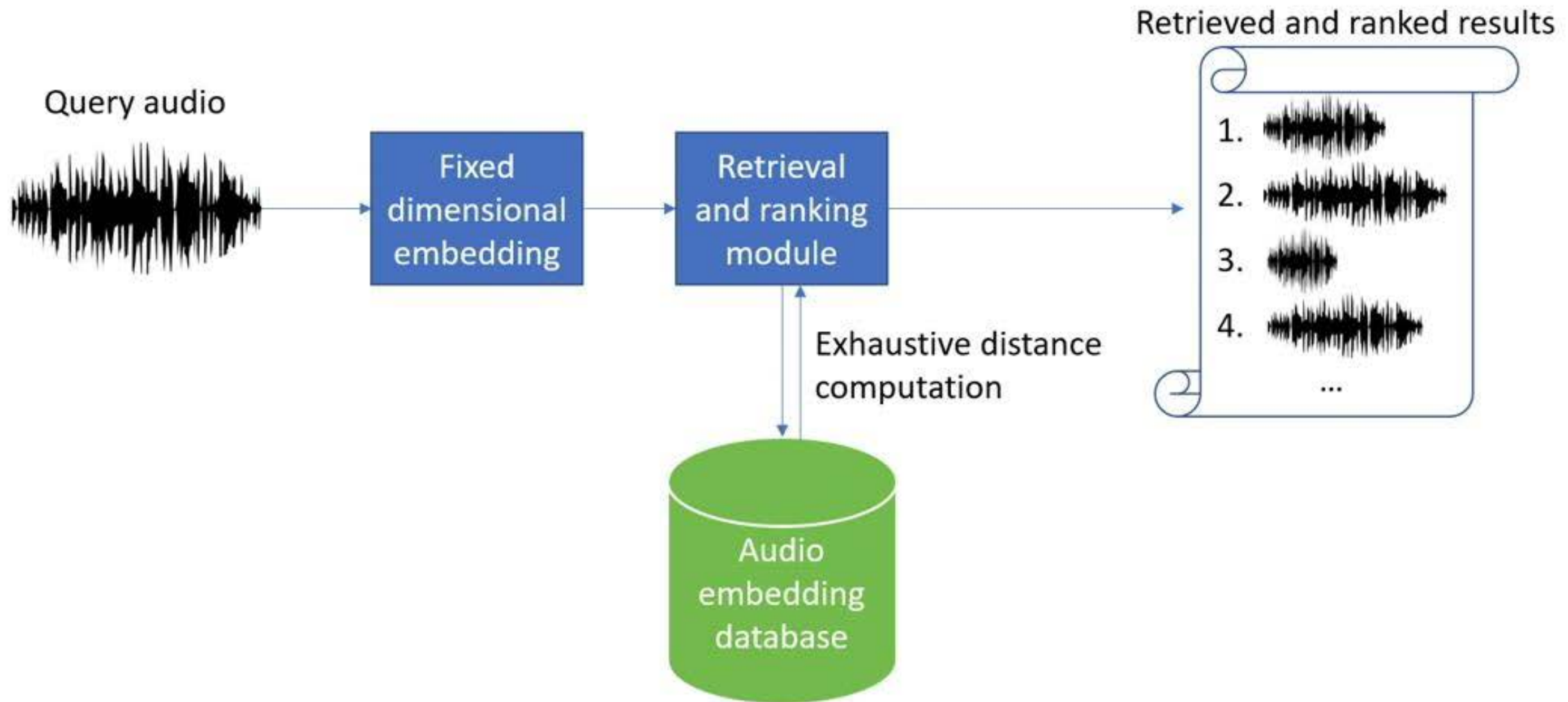
Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$O(1B)$	$O(256^8 \times 10^3)$ intractable	$O(8.3M)$?
Effective # codewords	--	K	$(K_{subspace})^M$?
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K ⇒ high error	Cannot retain data pattern in hash codes	Needs some labeled data

N = # samples in the database = 1M
 D = feature dimension = 1000
 M = # of subspaces = 8
 # centroids per subspace, $K_{subspace}$ = 256
 $\Rightarrow M \log_2 K_{subspace}$ = 64 bits hash code
 Effective K in PQ, $K = (K_{subspace})^M = 256^8$

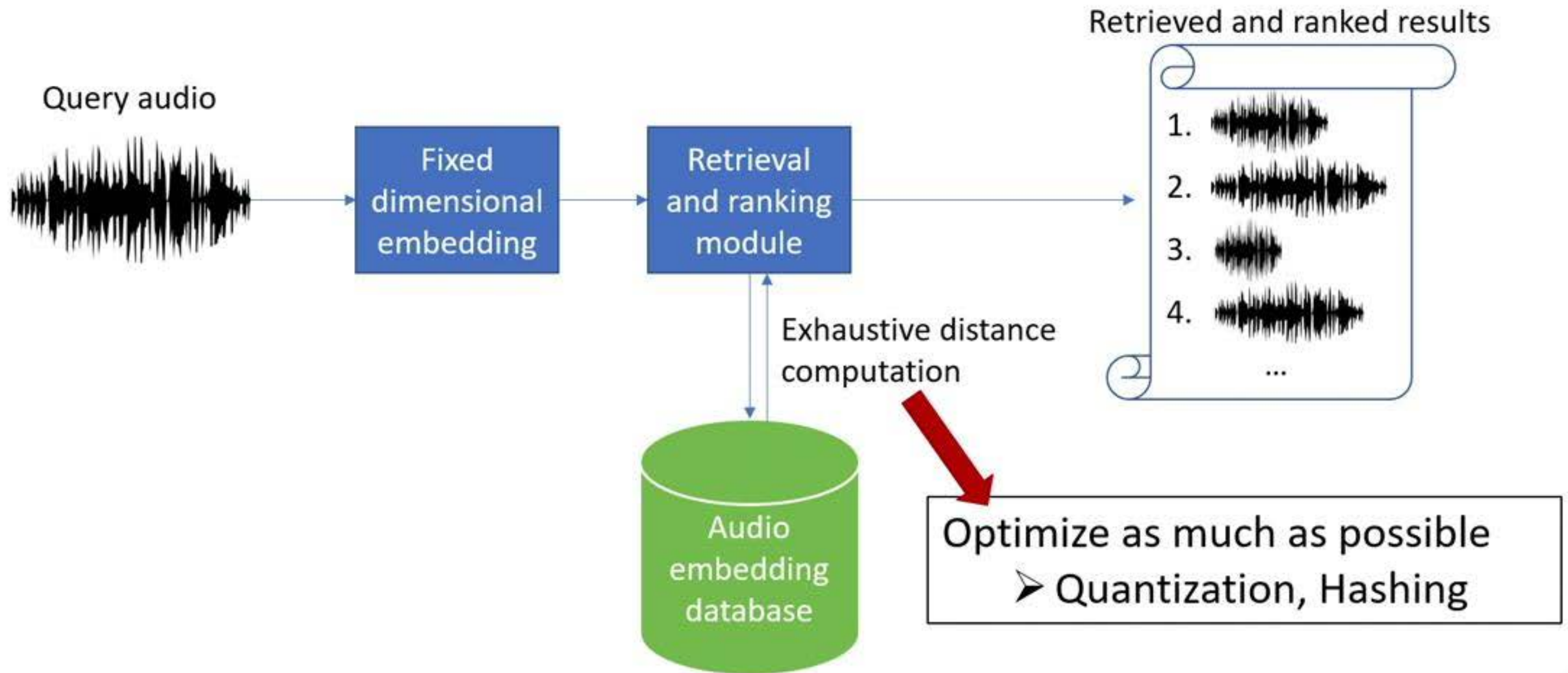
Efficient Audio Retrieval and Ranking

High-Level View- Recap



Efficient Audio Retrieval and Ranking

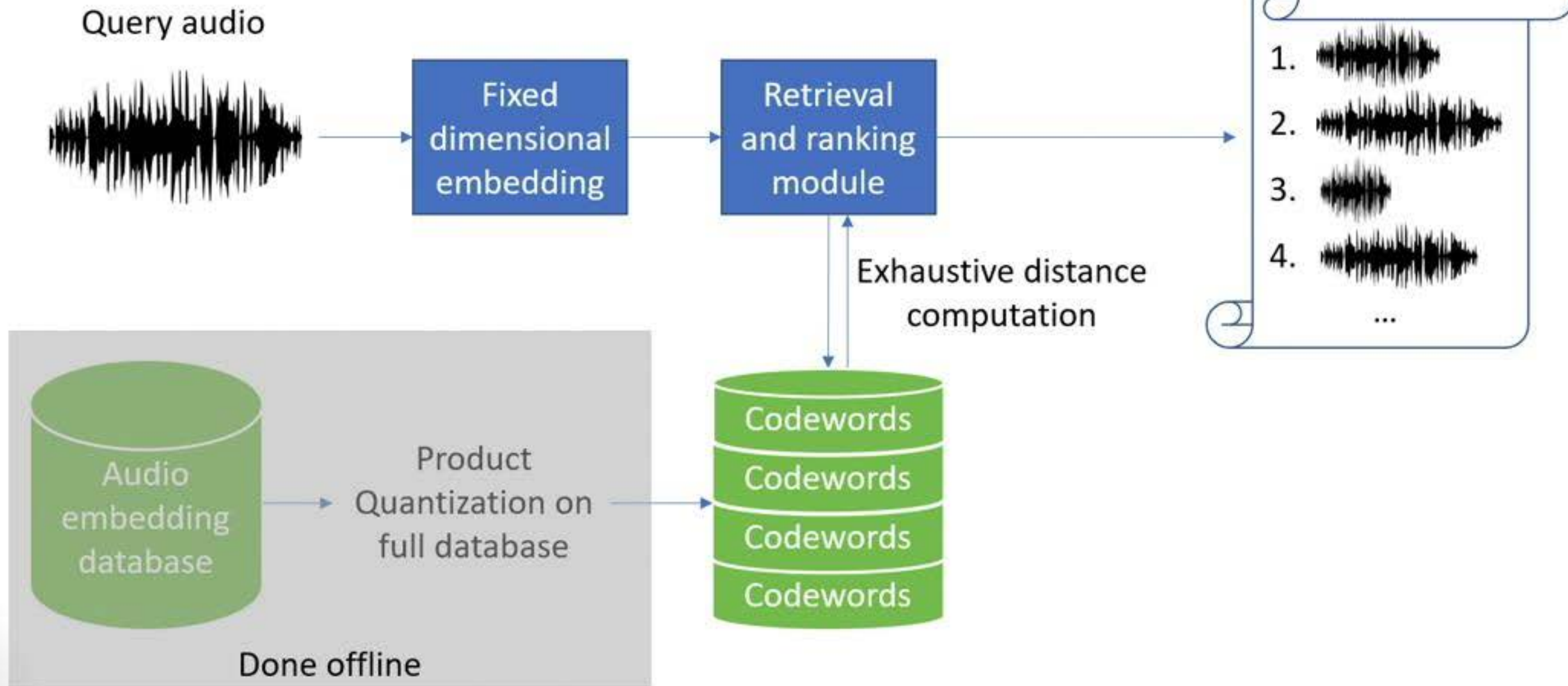
High-Level View- Recap



Efficient Audio Retrieval with Hashing

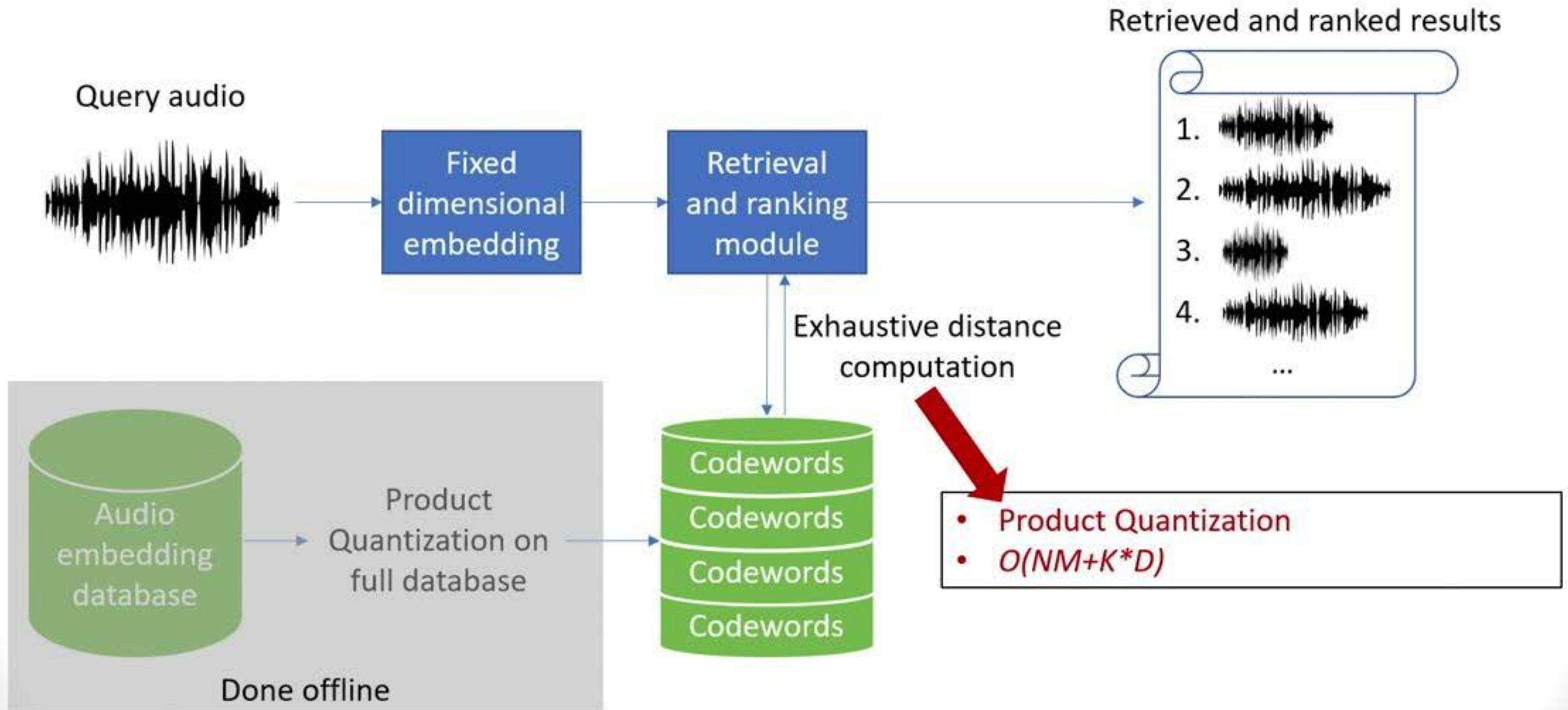
Store Hash Codes instead of float embeddings

Retrieved and ranked results



Efficient Audio Retrieval with Hashing

Store Hash Codes instead of float embeddings

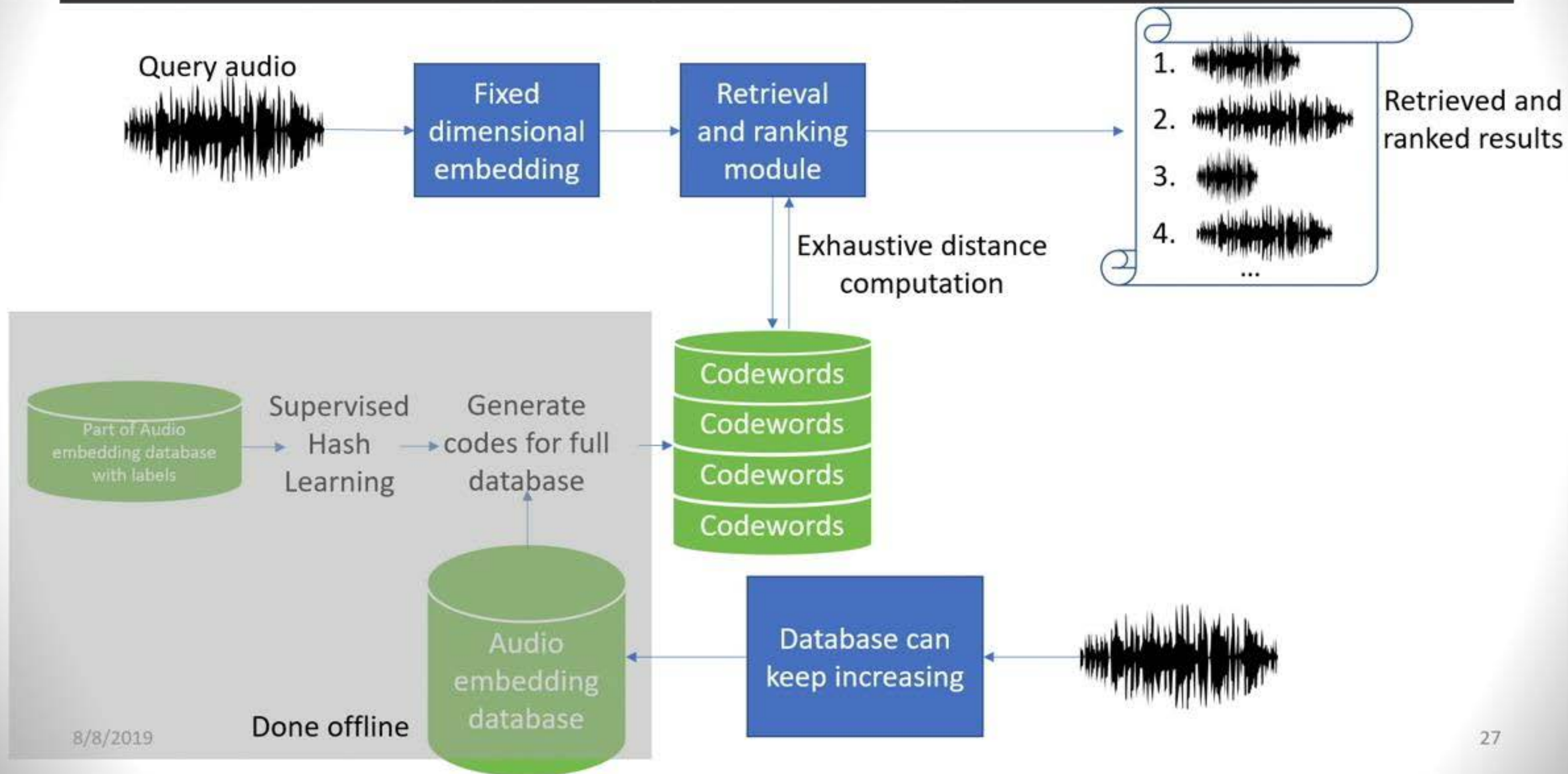


Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- **Efficient audio retrieval with hashing**
 - Unsupervised hashing algorithms
 - **Supervised deep hashing**
 - Experimental setting
 - Results
- Conclusions and future work

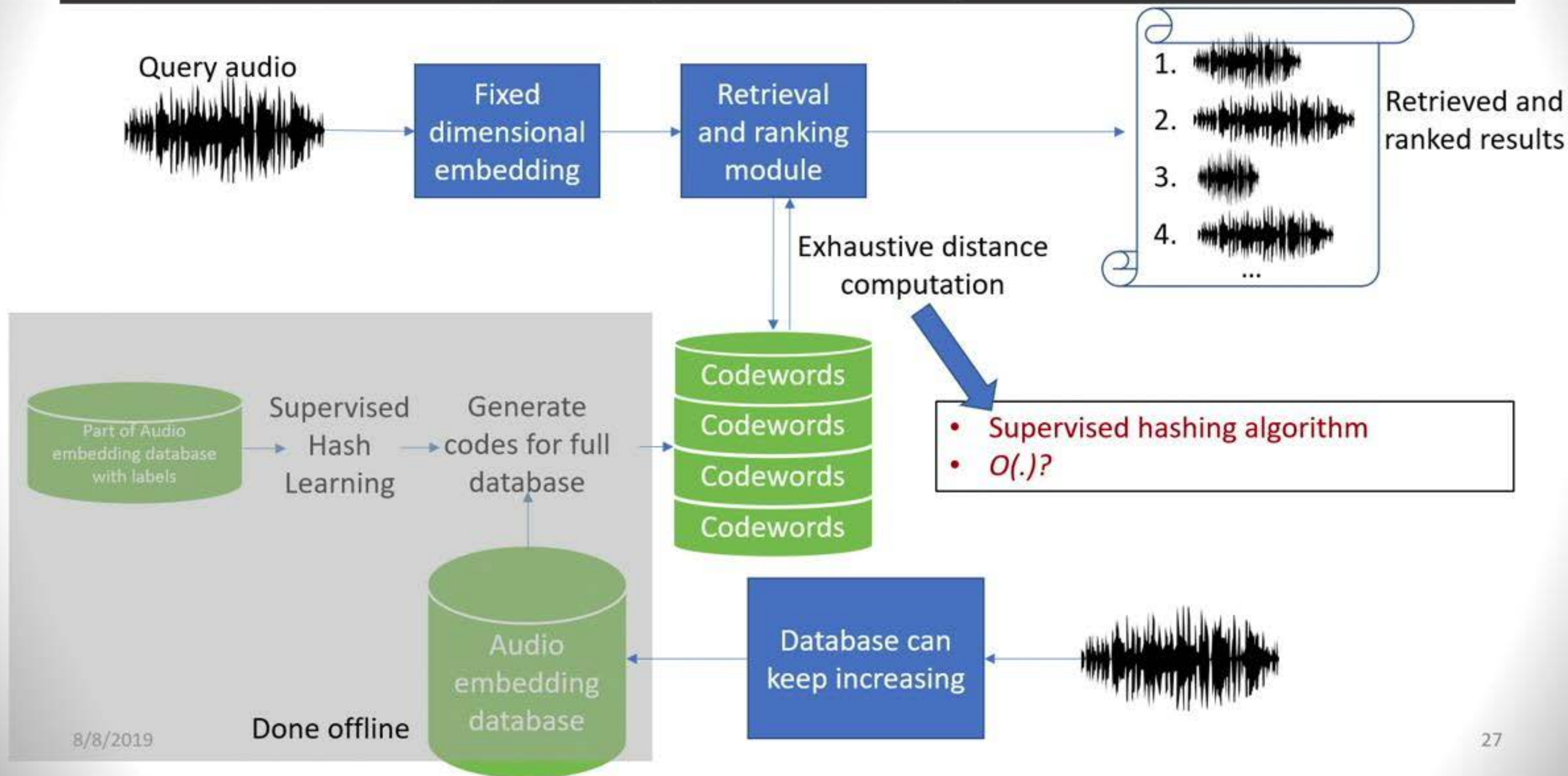
Efficient Audio Retrieval with Hashing

Paradigm of Supervised Hashing for Retrieval



Efficient Audio Retrieval with Hashing

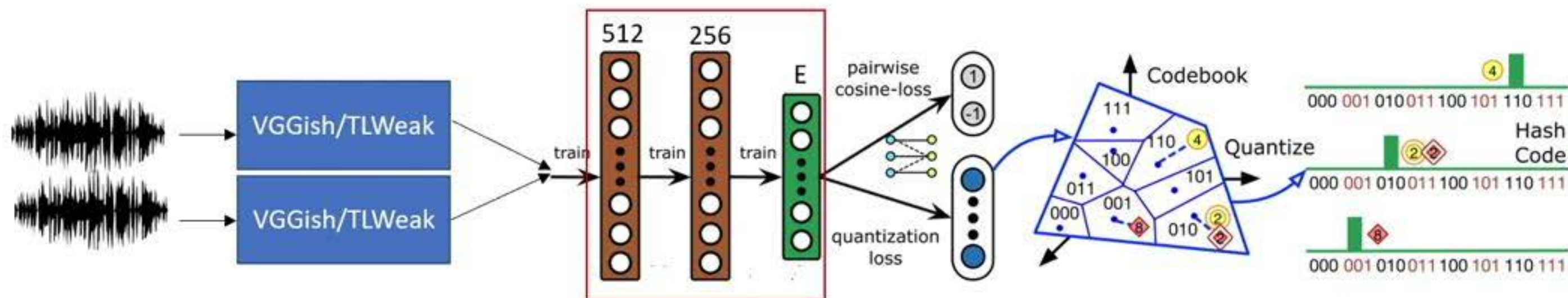
Paradigm of Supervised Hashing for Retrieval



Supervised Deep Hashing

Deep Quantization Network (DQN)

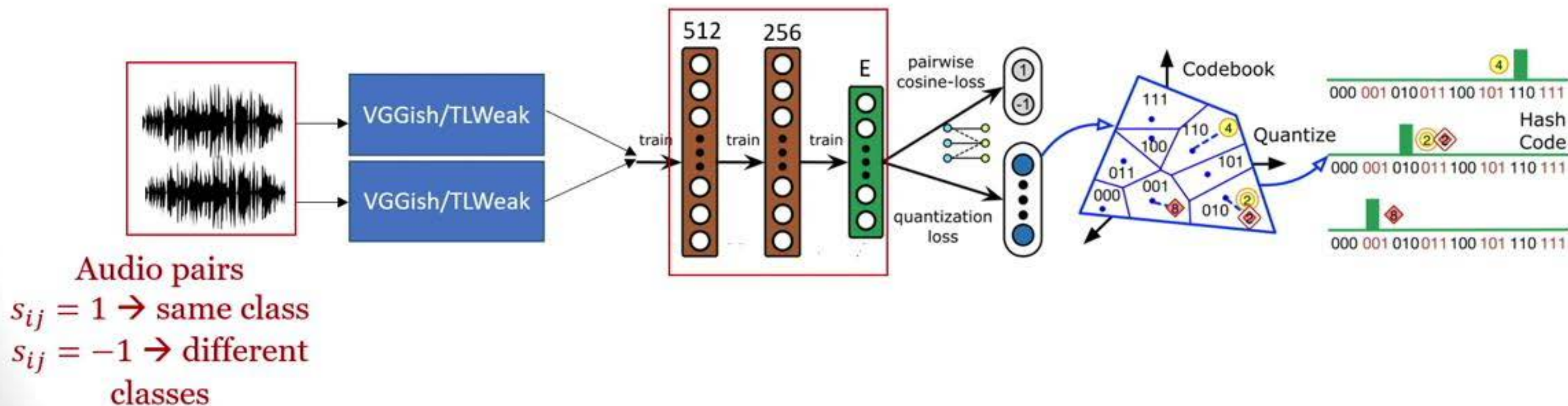
- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)



Supervised Deep Hashing

Deep Quantization Network (DQN)

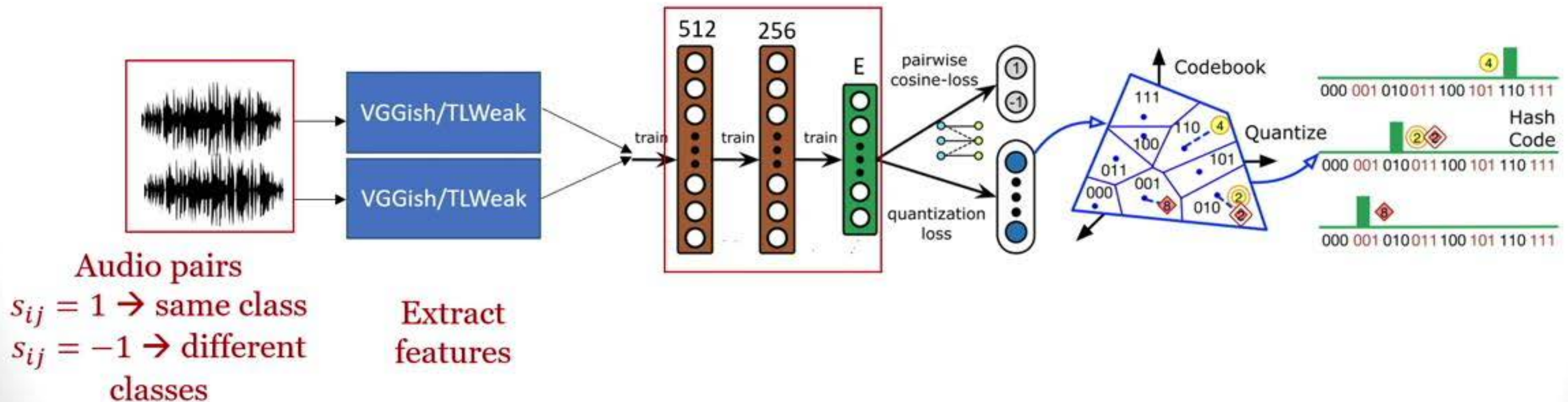
- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)



Supervised Deep Hashing

Deep Quantization Network (DQN)

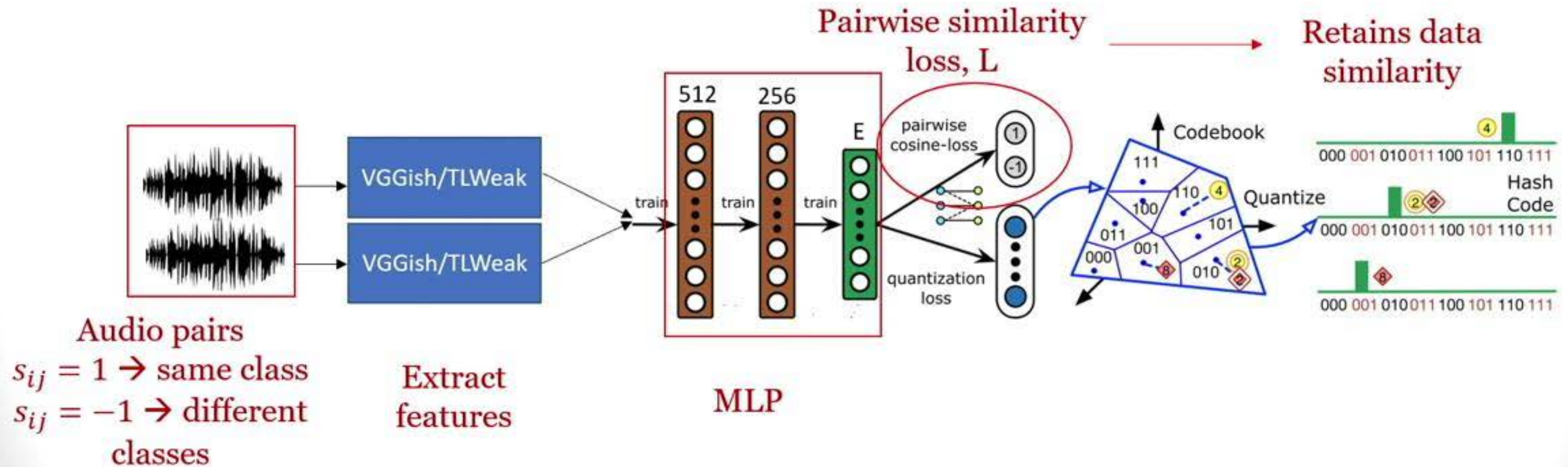
- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)



Supervised Deep Hashing

Deep Quantization Network (DQN)

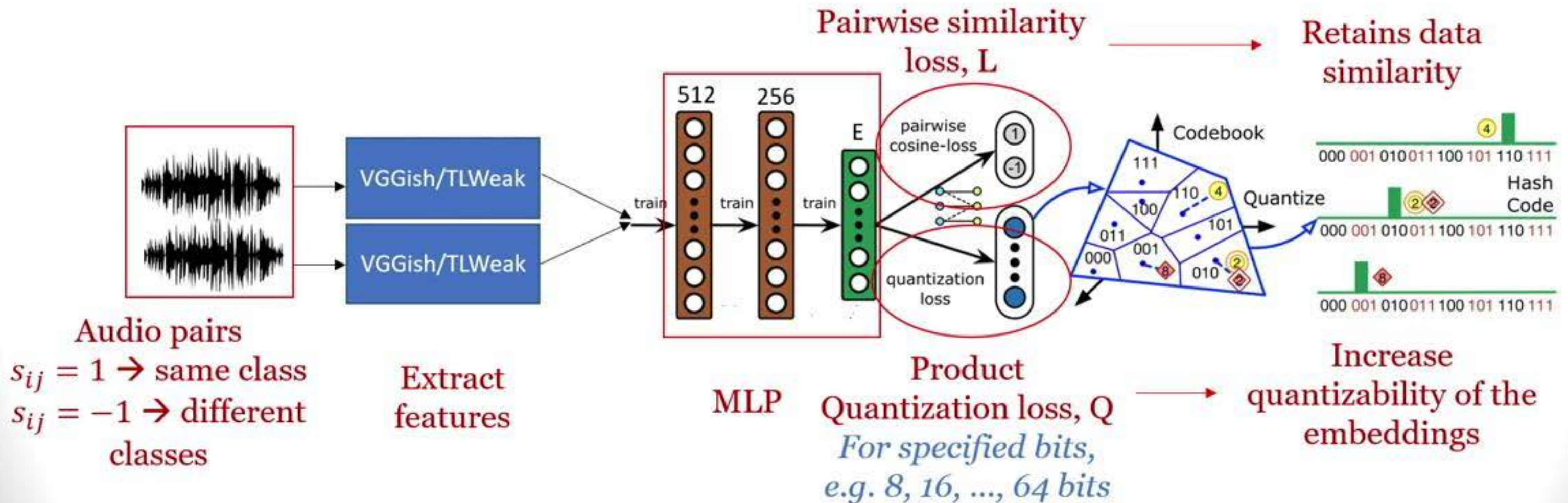
- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)



Supervised Deep Hashing

Deep Quantization Network (DQN)

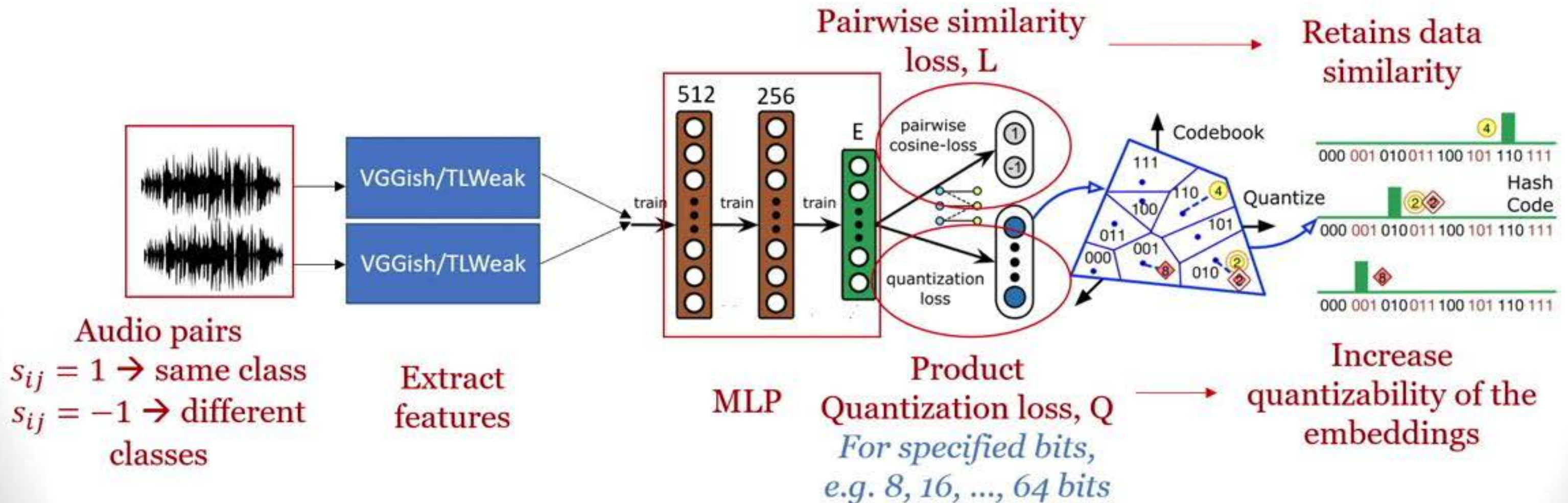
- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)



Supervised Deep Hashing

Deep Quantization Network (DQN)

- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)





- $Loss = L + \lambda Q$

Cao, Yue, Mingsheng Long, Jianmin Wang, Han Zhu, and Qingfu Wen. "Deep quantization network for efficient image retrieval." In Thirtieth AAAI Conference on Artificial Intelligence. 2016.

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

Comparison

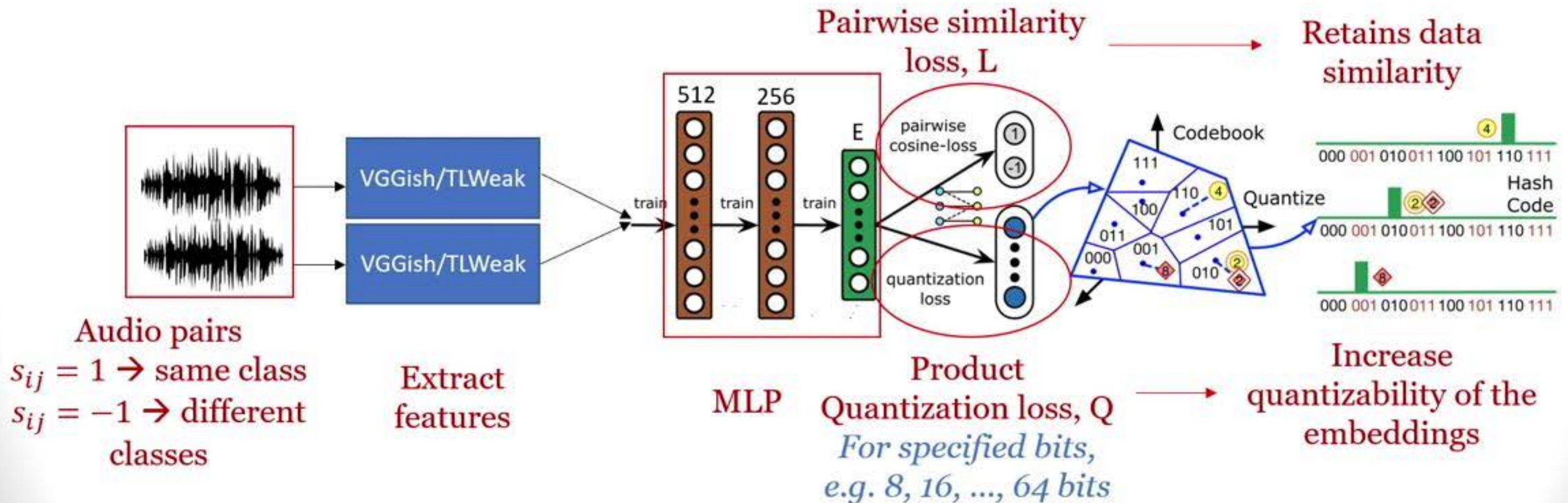
	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$	$\mathcal{O}(NM + K_{subspace}D + \mathbf{DNN}(\mathbf{x}))$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

$\mathbf{DNN}(\mathbf{x})$ = forward prop. for one sample

Supervised Deep Hashing

Deep Quantization Network (DQN)

- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)





- $\text{Loss} = L + \lambda Q$

Cao, Yue, Mingsheng Long, Jianmin Wang, Han Zhu, and Qingfu Wen. "Deep quantization network for efficient image retrieval." In Thirtieth AAAI Conference on Artificial Intelligence. 2016.

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN



Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$	$\mathcal{O}(NM + K_{subspace}D + \mathbf{DNN}(\mathbf{x}))$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

$\mathbf{DNN}(\mathbf{x})$ = forward prop. for one sample

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

Comparison



	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$	$\mathcal{O}(NM + K_{subspace}D + \mathbf{DNN}(\mathbf{x}))$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

N = # samples in the database = 1M
 D = feature dimension = 1000
 M = # of subspaces = 8
 # centroids per subspace, $K_{subspace}$ = 256
 $\Rightarrow M \log_2 K_{subspace}$ = 64 bits hash code
 Effective K in PQ, $K = (K_{subspace})^M = 256^8$

$\mathbf{DNN}(\mathbf{x})$ = forward prop. for one sample

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$O(1B)$	$O(256^8 \times 10^3)$ intractable	$O(8.3M)$	$O(9M)$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

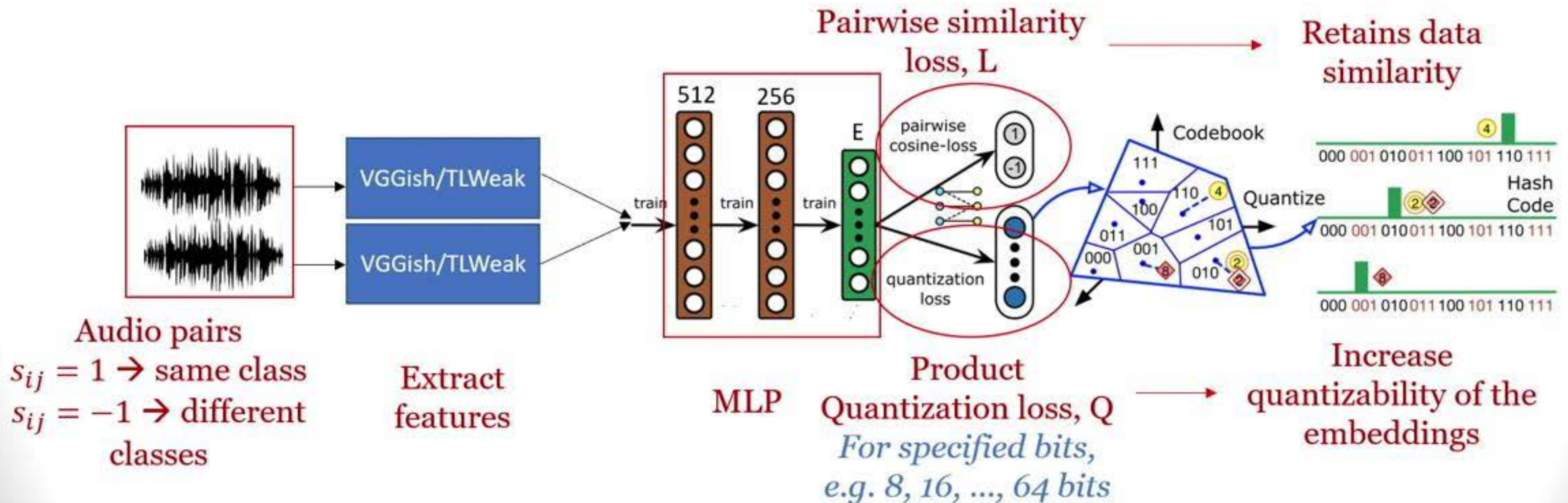
$N = \#$ samples in the database = 1M
 $D =$ feature dimension = 1000
 $M = \#$ of subspaces = 8
 $\#$ centroids per subspace, $K_{subspace} = 256$
 $\Rightarrow M \log_2 K_{subspace} = 64$ bits hash code
 Effective K in PQ, $K = (K_{subspace})^M = 256^8$

$DNN(x)$ = forward prop. for one sample

Supervised Deep Hashing

Deep Quantization Network (DQN)

- Contributions:
 - Combines feature learning and hashing together
 - Has a formal control over quantization error (earlier methods did not)





- $\text{Loss} = L + \lambda Q$

Cao, Yue, Mingsheng Long, Jianmin Wang, Han Zhu, and Qingfu Wen. "Deep quantization network for efficient image retrieval." In Thirtieth AAAI Conference on Artificial Intelligence. 2016.

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN



Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$\mathcal{O}(ND)$	$\mathcal{O}(KD)$	$\mathcal{O}(NM + K_{subspace}D)$	$\mathcal{O}(NM + K_{subspace}D + \mathbf{DNN}(\mathbf{x}))$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

$\mathbf{DNN}(\mathbf{x})$ = forward prop. for one sample

Euclidean \rightarrow VQ \rightarrow PQ \rightarrow DQN

Comparison

	Euclidean	VQ	PQ	DQN
Exhaustive distance computation complexity	$O(1B)$	$O(256^8 \times 10^3)$ intractable	$O(8.3M)$	$O(9M)$
Effective # codewords	--	K	$(K_{subspace})^M$	$(K_{subspace})^M$
 Pros	Most accurate	Simple	Supports exponentially large number of codewords	Retains data similarity
 Cons	Expensive	Cannot support exponentially large K \Rightarrow high error	Cannot retain data pattern in hash codes	Needs some labeled data

$N = \#$ samples in the database = 1M
 $D =$ feature dimension = 1000
 $M = \#$ of subspaces = 8
 $\#$ centroids per subspace, $K_{subspace} = 256$
 $\Rightarrow M \log_2 K_{subspace} = 64$ bits hash code
 Effective K in PQ, $K = (K_{subspace})^M = 256^8$

$DNN(x)$ = forward prop. for one sample

Experimental Setting

Datasets

- DCASE 2018 Task-2:

- Test: 1600 audio files
- Train: 9473 audio files
- Number of audio classes: 41



Experimental Setting

Datasets

- DCASE 2018 Task-2:

- Test: 1600 audio files
- Train: 9473 audio files
- Number of audio classes: 41



acoustic_guitar



Experimental Setting

Datasets

- DCASE 2018 Task-2:

- Test: 1600 audio files
- Train: 9473 audio files
- Number of audio classes: 41



acoustic_guitar



Cello



Scissors



Experimental Setting

Datasets

- DCASE 2018 Task-2:

- Test: 1600 audio files
- Train: 9473 audio files
- Number of audio classes: 41



acoustic_guitar



Cello



Scissors



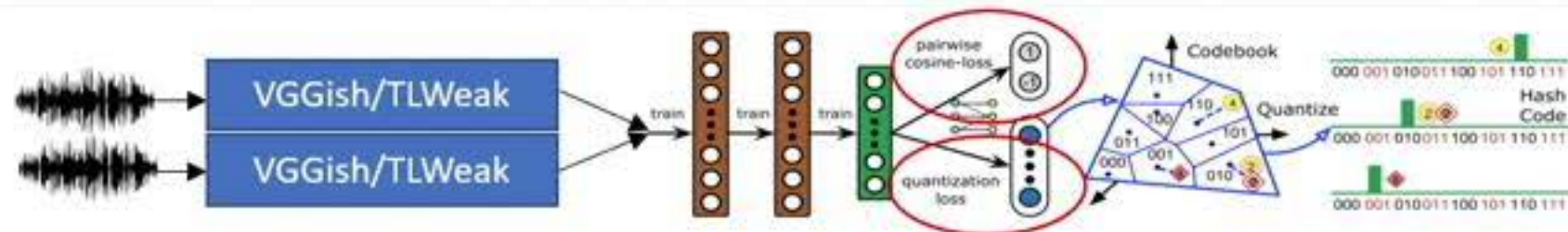
Cowbell

- ESC-50:

- Test: 400
- Train: 1600
- Number of audio classes: 50

Experimental Setting

Features / pretrained embeddings



- VGGish:
 - Network: Deep CNN, VGG
 - Feature/embedding Dimension: **128**
 - Training data: *Weakly labeled* 70M training videos (5.24 million hours)!
- TLWeak:
 - Network: Deep CNN
 - Feature/embedding Dimension: **1024**
 - Training data: Google's AudioSet, balanced training
 - State-of-the-art on AudioSet

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

Evaluation

Mean Average Precision (mAP@R)

- **mean Average Precision@R:**

Evaluation

Mean Average Precision (mAP@R)

- **mean Average Precision@R:**

- Number of retrieved items
- Different applications might have different requirement

Evaluation

Mean Average Precision (mAP@R)

- mean **Average Precision@R:**

- % of positive retrievals that are correct

- Number of retrieved items
- Different applications might have different requirement

Evaluation

Mean Average Precision (mAP@R)

- **mean Average Precision@R:**

- Mean over all queries

- % of positive retrievals that are correct

- Number of retrieved items
- Different applications might have different requirement

- Properties:

- $0 \leq \text{mAP@R} \leq 1$

- Depends on ranking/ordering of the retrieved samples

Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- **Efficient audio retrieval with hashing**
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - **Results**
- Conclusions and future work

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

DCASE Results, mAP@database_size

Comparison between different algorithms

TLWeak 1024

Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
	ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
	AGH	9.40%	13.03%	15.13%	15.35%	16.49%
	PQ	15.06%	16.15%	16.30%	16.39%	16.36%
Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%

VGG 128

Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
	ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
	AGH	13.37%	15.04%	15.52%	16.34%	15.41%
	PQ	16.12%	16.34%	16.23%	16.24%	15.65%
Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
		Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%
	VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%
ITQ			10.10%	12.18%	14.25%	14.61%	15.64%
AGH			13.37%	15.04%	15.52%	16.34%	15.41%
PQ			16.12%	16.34%	16.23%	16.24%	15.65%
Supervised (~10% of database for training)			DQN	33.84%	38.93%	39.68%	40.31%

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
	VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%
ITQ			10.10%	12.18%	14.25%	14.61%	15.64%
AGH			13.37%	15.04%	15.52%	16.34%	15.41%
PQ			16.12%	16.34%	16.23%	16.24%	15.65%
Supervised (~10% of database for training)		DQN	33.84%	38.93%	39.68%	40.31%	41.43%

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

DCASE Results, mAP@database_size

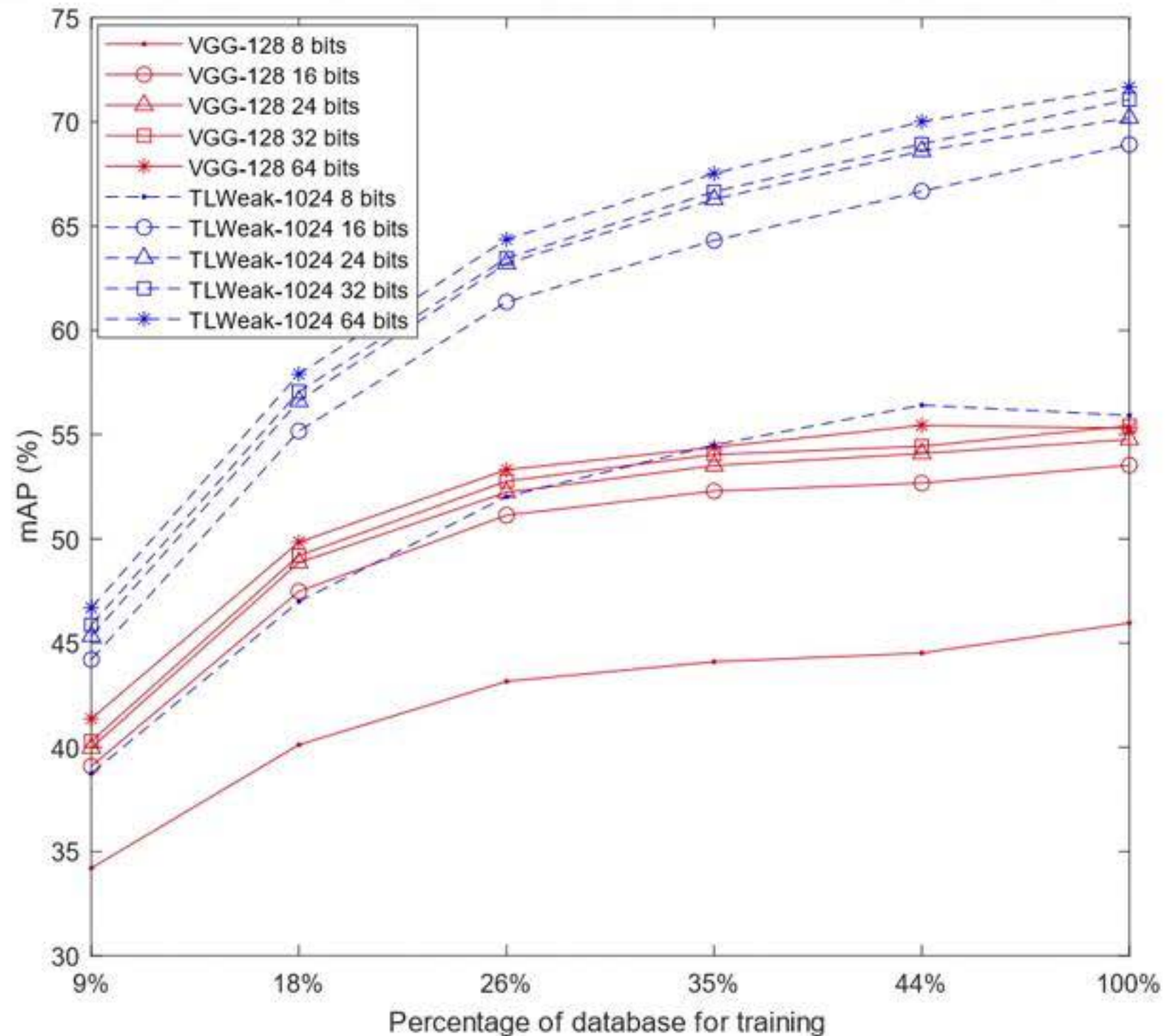
Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

Similar findings in ESC-50 dataset

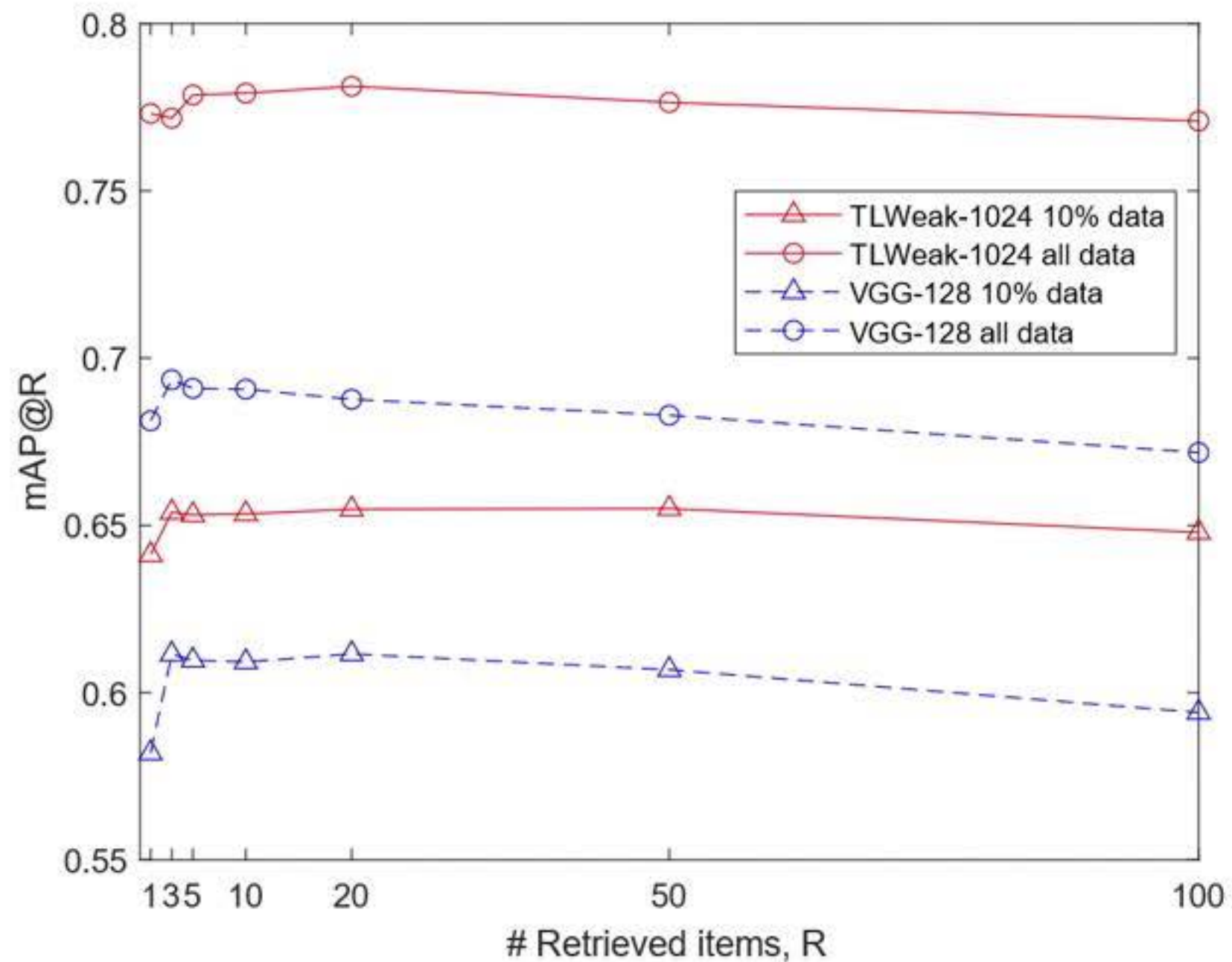
DCASE Results, mAP@database_size

Vary training dataset size



DCASE Results

mAP@R for different # retrieved items



Agenda

- Audio event detection & classification
- Audio retrieval and ranking
 - Literature review
- Efficient audio retrieval with hashing
 - Unsupervised hashing algorithms
 - Supervised deep hashing
 - Experimental setting
 - Results
- **Conclusions and future work**

Conclusions and Future Works

Suggestions are welcome!

- Contributions:
 - First attempt for efficient audio retrieval
 - Saves millions of operations in nearest neighbor search
 - Small amount of labeled data can boost the performance by absolute 30%
 - Validated on multiple datasets and features
- Future works:
 - Non-exhaustive search for even faster retrieval
 - **“The Inverted Multi-Index”** algorithm (Babenko et. al.)
 - Hashing for cross-modal retrieval (Elizalde et. al.)
 - Hierarchical audio event hashing and retrieval
 - Hash codes that preserves ontology information

Babenko and V.S. Lempitsky, "The Inverted Multi-Index," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 3069-3076, 2012.

Elizalde, Benjamin, Shuayb Zarar, and Bhiksha Raj. "Cross Modal Audio Search and Retrieval with Joint Embeddings Based on Text and Audio." IEEE ICASSP 2019.



DCASE Results, mAP@database_size

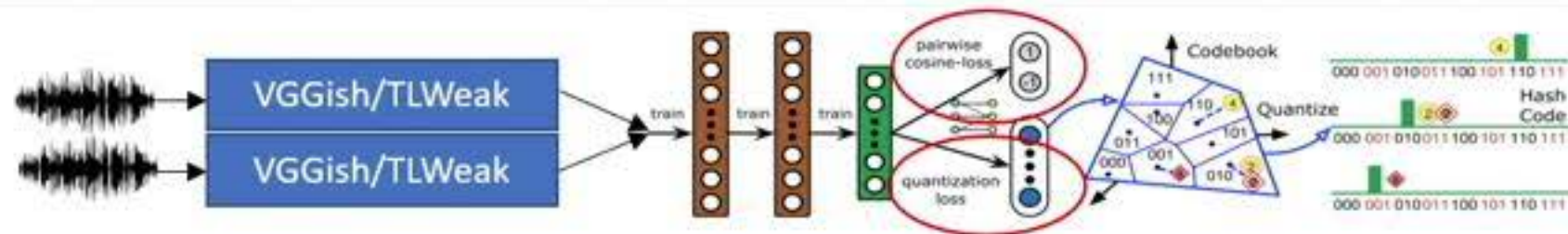
Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%

Similar findings in ESC-50 dataset

Experimental Setting

Features / pretrained embeddings



- VGGish:
 - Network: Deep CNN, VGG
 - Feature/embedding Dimension: **128**
 - Training data: *Weakly labeled* 70M training videos (5.24 million hours)!
- TLWeak:
 - Network: Deep CNN
 - Feature/embedding Dimension: **1024**
 - Training data: Google's AudioSet, balanced training
 - State-of-the-art on AudioSet

VGGish: Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." IEEE ICASSP, 2017.

TLWeak: Kumar, Anurag, Maksim Khadkevich, and Christian Fügen. "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes." IEEE ICASSP, 2018.

DCASE Results, mAP@database_size

Comparison between different algorithms

	Training	Algorithm	bit_8	bit_16	bit_24	bit_32	bit_64
TLWeak 1024	Unsupervised (Full database for training)	SH	8.14%	10.97%	11.93%	12.84%	13.31%
		ITQ	8.34%	12.03%	14.17%	15.52%	17.62%
		AGH	9.40%	13.03%	15.13%	15.35%	16.49%
		PQ	15.06%	16.15%	16.30%	16.39%	16.36%
	Supervised (~10% of database for training)	DQN	39.07%	44.24%	45.50%	45.77%	46.83%
VGG 128	Unsupervised (Full database for training)	SH	9.23%	11.34%	12.18%	12.85%	12.90%
		ITQ	10.10%	12.18%	14.25%	14.61%	15.64%
		AGH	13.37%	15.04%	15.52%	16.34%	15.41%
		PQ	16.12%	16.34%	16.23%	16.24%	15.65%
	Supervised (~10% of database for training)	DQN	33.84%	38.93%	39.68%	40.31%	41.43%